Common Criteria for Information Technology Security Evaluation

---

CCEB-96/012_A

Part 2 : Annexes

Version 1.0

96/01/31

Foreword

Following extensive international cooperation to align the source criteria from Canada (CTCPEC), Europe (ITSEC) and the United States of America (TCSEC and Federal Criteria), version 1.0 of the *Common Criteria for Information Technology Security Evaluation* is issued for the purpose of trial evaluations and for review by the international security community. The practical experience acquired through trial evaluations and all the comments received will be used to further develop the criteria.

A template for reporting observations on version 1.0 of the CC is included at the end of the annexes of this document. Any observation reports should be communicated to one or more of the following points of contact at the sponsoring organisations:

National Institute of Standards and Technology
Computer Security Division
NIST North Building, Room 426
Gaithersburg, Maryland 20899
U.S.A.
Tel: (+1)(301)975-2934, Fax:(+1)(301)926-2733
E-mail:csd@nist.gov
http://csrc.ncsl.nist.gov

National Security Agency
Attn: V2, Common Criteria Technical Advisor
Fort George G. Meade, Maryland 21122
U.S.A.
Tel: (+1)(410)859-4458, Fax:(+1)(410)684-7512
E-mail: common_criteria@radium.ncsc.mil

Communications Security Establishment
Criteria Coordinator
R2B IT Security Standards and Initiatives
P.O. Box 9703, Terminal
Ottawa, Canada K1G 3Z4
Tel:(+1)(613)991-7409, Fax:(+1)(613)991-7411
E-mail:criteria@cse.dnd.ca
ftp:ftp.cse.dnd.ca
http://www.cse.dnd.ca

UK IT Security and Certification Scheme
Senior Executive
P.O. Box 152
Cheltenham GL52 5UF
United Kingdom
Tel: (+44) 1242 235739, Fax:(+44)1242 235233
E-mail: ccv1.0@itsec.gov.uk
ftp: ftp.itsec.gov.uk
http://www.itsec.gov.uk

Bundesamt für Sicherheit in der Informationstechnik
Abteilung V
Postfach 20 03 63
D-53133 Bonn
Germany
Tel: (+49)228 9582 300, Fax:(+49)228 9582 427
E-mail:cc@bsi.de

Service Central de la Sécurité des Systèmes d'Information
Bureau Normalisation, Critères Communs
18 rue du docteur Zamenhof
92131 Issy les Moulineaux
France
Tel: (+33)(1)41463784, Fax:(+33)(1)41463701
E-mail:ssi28@calvacom.fr

Netherlands National Communications Security Agency
P.O. Box 20061
NL 2500 EB The Hague
The Netherlands
Tel: (+31) 70 3485637, Fax:(+31).70.3486503
E-mail: criteria@nlncsa.minbuza.nl

This document is paginated from i to x and from 1 to 246

Table of contents

List of figures

List of tables

Chapter 1

Security functional requirements application notes

1 This Annex contains informative guidance for the families and components found in the main body of Part 2 which may be required by users, developers or evaluators to use the components. To facilitate finding the appropriate information, the presentation of the classes, families, and components in this Annex are similar to the main body of Part 2. The class, family, and component structures in this annex differ from that found in the main body of Part 2 since this Annex is concerned with only those sections which are informative.

1.1 Overview

2 This section defines the content and presentation of the notes related to functional requirements of the CC. It provides guidance on the organisation of the requirements for the supporting information provided for new components to be included in a security target and to be evaluated.

1.1.1 Class structure

3 Figure 1.1 below illustrates the functional class structure in this annex in diagrammatic form.

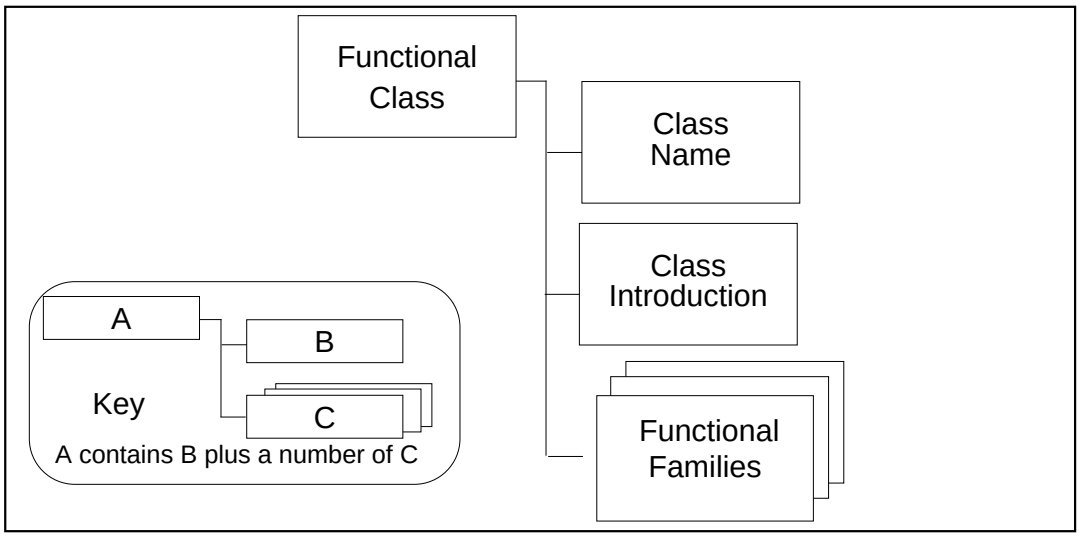

Figure 1.1 - Functional class structure.

1.1.1.1 Class name

4 This is the unique name of the class defined in Part 2 of the CC.

1.1.1.2 Class introduction

5 The class introduction in this annex provides information about the construction rules to use families and components of the class to set up a consistent PP, ST, or functional packages. This information is completed with the informative diagram that describes the organisation of each class with the families in each class and the hierarchical relationship between components in each family.

1.1.2 Family structure

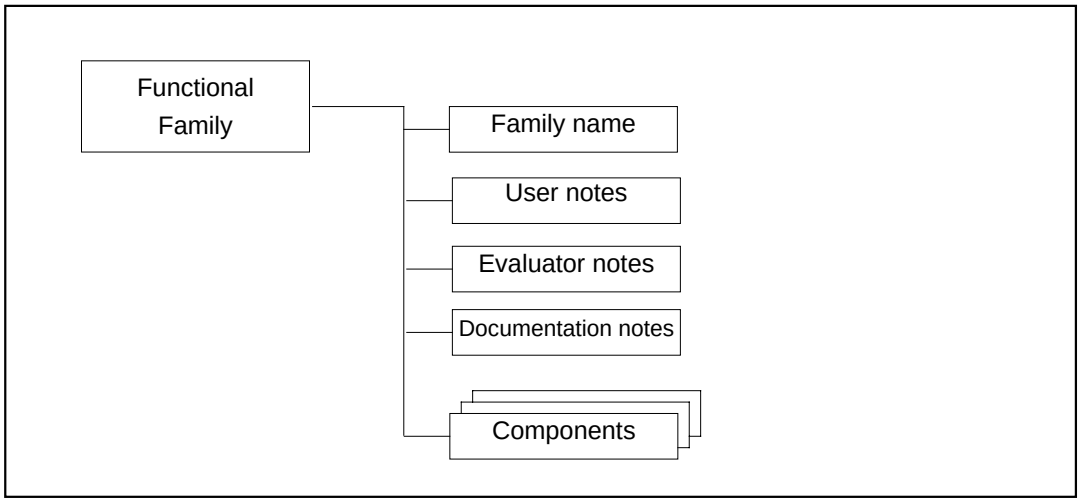

Figure 1.2 - Functional family structure for Application notes

6 Figure 1.2 illustrates the functional family structure for application notes in diagrammatic form.

1.1.2.1 Family name

7 This is the unique name of the family defined in Part 2 of the CC.

1.1.2.2 User notes

8 The *user notes* contain additional information which is of interest to potential users of the family, that is, PP, ST, and functional package authors and developers of TOEs incorporating the functional components. The presentation is informative and might cover, for example, warnings about limitations of use and areas where specific attention might be required when using the components.

1.1.2.3 Evaluator notes

9 The *evaluator notes* contain any information that is of interest to developers and evaluators of TOEs that claim compliance to a component of the family. The presentation is informative and can cover a variety of areas where specific attention might be needed when evaluating the TOE. This can include what needs to be documented to support the required functional behaviour, clarifications of meaning and specification of the way to interpret specific requirements, as well as caveats and warnings of specific interest to evaluators.

1.1.2.4 Documentation notes

10 The *documentation notes* contain information that may be of interest to PP/ST authors when defining the set of expected information to be provided by the relevant documentation, part of the evaluation deliverables. The presentation is informative and is in the form of suggestions that are not considered as normative on the part of PP/ST authors.

11 These note sections are not mandatory and should appear only if appropriate.

1.1.3 Component structure

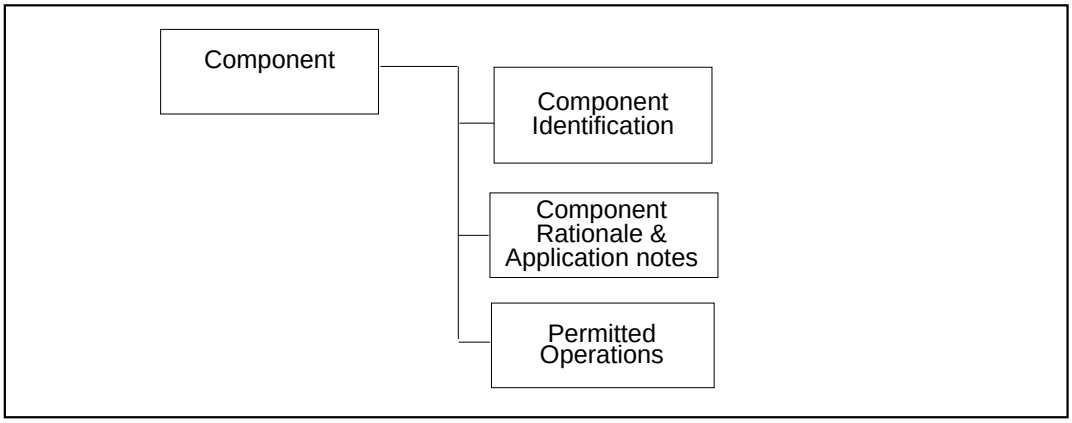

Figure 1.3 - Functional component structure

12 Figure 1.3 illustrates the required functional component structure for the application notes.

1.1.3.1 Component identification

13 This is the unique name of the component defined in Part 2 of the CC.

1.1.3.2 Component rationale and application notes

14 Any specific information related to the component should be provided in this section to enhance the description of the application notes defined in the family.

- The *rationale* contains the specifics of the rationale that refines the general statements on rationale for the specific level, and should only be used if level specific amplification is required.

- The *application notes* contains additional refinement in terms of narrative qualification as it pertains to a specific component. This refinement can pertain to user notes, evaluator notes, and/or documentation notes as described in section 1.1.2 of this annex. This refinement can be used to explain dependencies (e.g. shared information, or shared operation).

15 This section is not mandatory and should appear only if appropriate.

1.1.3.3 Permitted operations

16 Components may be tailored through use of permitted operations before being incorporated into a PP, an ST, or a functional package, based on the particular environment of use and security policies being addressed. The possible operations are defined in the CC Part 2 document and elaborated on in this Annex. Not all operations are permitted on all functional components. Each component shall contain a description of the allowed operations, the circumstances under which the operation can be applied to the component, and the results of the application of this operation.

17 This section is not mandatory and should appear only if appropriate.

1.2 TSF attribute requirements

18 The CC Part 2 functional components are built around several types of entities and attributes and describe functional requirements operating on these entities and attributes. In Table 1.1 - TSF Attribute Requirements, the functional families that provide requirements for management activities related to attributes are presented.

Table 1.1 - TSF Attribute Requirements

	User Data	TSF Data	Authentication Data	Security Attributes	User Attributes	Object Attributes	Subject Attributes
Definition[a]							
Initialisation			FIA_ADA		FIA_ATA	FDP_SAI[b]	FIA_USB[c]
Storage	FDP_ACC FDP_IFC	FPT_SWM	FIA_ADP		FIA_ATP		
Modification			FIA_ADA		FIA_ATA	FDP_SAM	
Query			FIA_ADA		FIA_ATA	FDP_SAQ	
Deletion[d]	FDP_RIP				FIA_ATA	FDP_SAM	

Table 1.1 - TSF Attribute Requirements

	User Data	TSF Data	Authentication Data	Security Attributes	User Attributes	Object Attributes	Subject Attributes
Transfer	FDP_UTC FDP_UTI	FPT_ITA FPT_ITC FPT_ITI		FDP_SAT			
Transfer Policy	FDP_ETC FDP_ITC[e]	FTP_ITC FTP_ITP					
Consistency				FDP_SAC[f]	FIA_ATC[g]		
Binding					FIA_ATD FIA_USB[h]	[i]	FIA_USB

a. Refinement in the PP/ST will give these definitions.
b. FDP_SAI is renamed to FDP Object Attributes Initialisation
c. During the binding of a subject to the user attributes the subject attributes will be initialised.
d. All functions capable of modifying security attributes are also capable of deleting security attributes.
e. Text refers to user information and general information.
f. This class is being shifted to FPT.
g. This class will be removed.
h. ATD: user to user attribute; USB user attributes to subjects
i. During the creation of an object the object attributes are being initialised.

Class FAU

Security Audit

19 CC audit families allow PP/ST authors the ability to define requirements for monitoring user activities and, in some cases, detecting real, potential, or imminent violations of the TSP. The TOE's security audit functions are defined to help monitor the use of access rights by all users, and act as a deterrent against security violations. The requirements of the audit families refer to functions that include audit data protection, record format, and event selection, as well as analysis tools, violation alarms, and real-time analysis. Audit data should be available in a useful format, that presents audit data in a human-readable format and/or delivers it to authorised users or processes acting on their behalf.

20 While developing the security audit requirements, the PP/ST author should take note of the inter-relationships among the audit families and components. The potential exists to specify a set of audit requirements that comply with the family/ component dependencies lists, while at the same time resulting in a deficient audit function (e.g., an audit function that requires all security relevant events to be audited but without the selectivity to control them on any reasonable basis such as individual user or object).

21 Figure 1.4 introduces construction rules for the audit requirements.

Construction rules

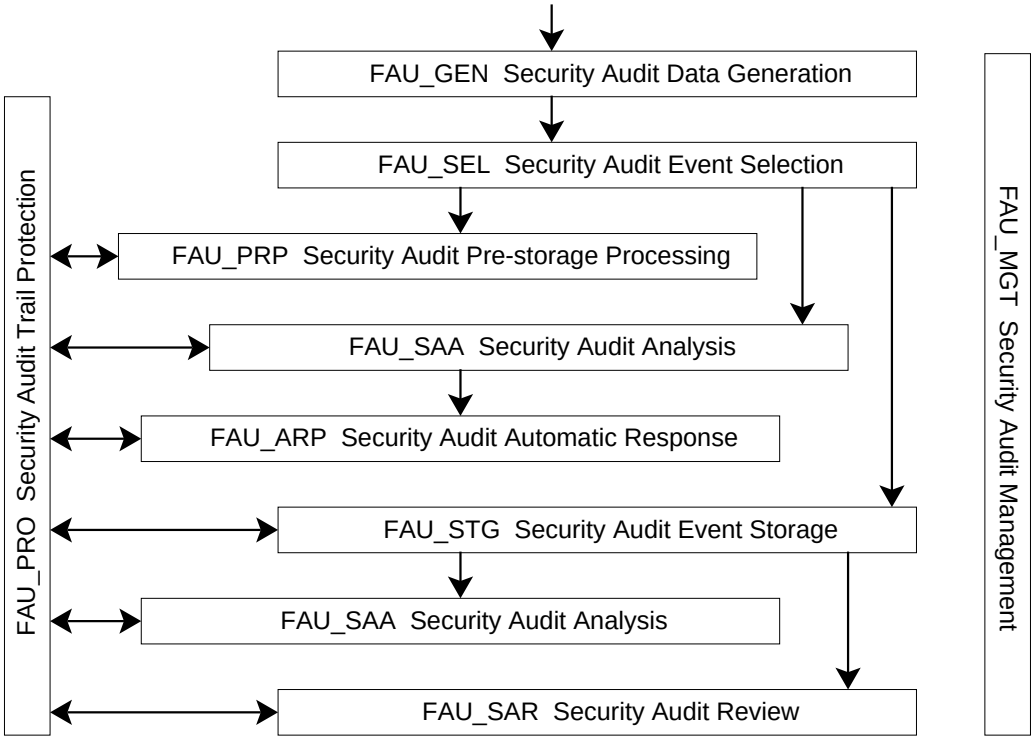

Figure 1.4 - Audit requirements construction rules

22 When building a PP, ST, or functional package using components from the FAU class, these construction rules provide guidance on where to look and what to select from the class.

23 When selecting a component in the FAU_GEN family, the PP/ST author defines:

- the level of granularity of auditable events;
- the list of data that should be recorded for those events; and
- the level of "user identity" (e.g., group or individual identification) for accounting those events.

24 When selecting a component in the FAU_MGT family, the PP/ST author requests:

- management function for the audit trail;
- control of the audit trail saturation; and
- management function for the audit trail saturation.

25 When selecting a component in the FAU_SEL family, the PP/ST author defines:

- the configuration mode (off-line or on-line);

- any authorised administrator ability to define the filter for auditing (based on object, user, subject attributes and/or auditable event types);
- any request for a runtime display facility; and
- the ability for authorised users to display selected auditable event types.

26 When selecting a component in the FAU_PRP family, the PP/ST author requests:

- processing of the audit record, to generate a “human readable” format, or an “automated treatment” format from audit data; and
- the rearranging of the audit data.

27 When selecting a component in the FAU_SAA family, the PP/ST author defines:

- the relevant rules for “Imminent violation detection”; and
- the ability for the authorised administrator to modify those rules.

28 When selecting a component in the FAU_ARP family, the PP/ST author defines:

- the complexity of the automatic response (alarm versus corrective action);
- if relevant: the least disruptive actions to be taken; and
- the ability for authorised administrator to modify those actions.

29 When selecting a component in the FAU_STG family, the PP/ST author requests:

- the storage of the audit record in an audit trail; and
- the prevention of audit records lost due to system failure, audit storage exhaustion or attack.

30 When selecting a component in the FAU_POP family, the PP/ST author requests:

- processing of the audit trail, to generate a “human readable” format, or an “automated treatment” format from audit information; and
- the rearranging of this representation by the authorised administrator.

31 When selecting a component in the FAU_SAR family, the PP/ST author requests:

- tools to review the audit data;
- the ability for authorised users to review audit data; and
- tools to select the information to be reviewed, with single or multiple criteria.

32 When selecting a component in the FAU_PRO family, the PP /ST writer requests:

- protection of the audit trail; and
- the ability for authorised users to read the audit trail.

Audit requirements in a distributed environment:

33 The appropriate audit requirements for networks and other large systems differ significantly from those needed for stand-alone systems. Larger, more complex and active systems require greater restraint in collecting audit data, due to lowered feasibility of interpreting (or even storing) what gets collected. The traditional

notion of a time-sorted list or “trail” of audited events may not be applicable in a global asynchronous network with arbitrarily many events occurring at once.

34 For example, a network router may not have a clock, and if so it certainly will not be globally synchronised for the net, and if synchronised it will differ by some delta due to message delays. Thus, audit of event time may be unavailable or at best be in error by that delta.

35 Also, different hosts and servers on a distributed TOE may certainly have differing naming policies and values. Symbolic names presentation for audit review may require a net-wide convention to avoid redundancies and “name clashes.”

36 A multi-object audit repository, portions of which are accessible by a potentially wide variety of authorised users, may be required if audit repositories are to serve a useful function in distributed systems.

37 Finally, misuse of authority by administrators can be addressed by systematically avoiding local storage of audit data pertaining to administrator actions.

38 Figures 1.5 and 1.6 show the decomposition of this class into its constituent components.

Component Catalogue

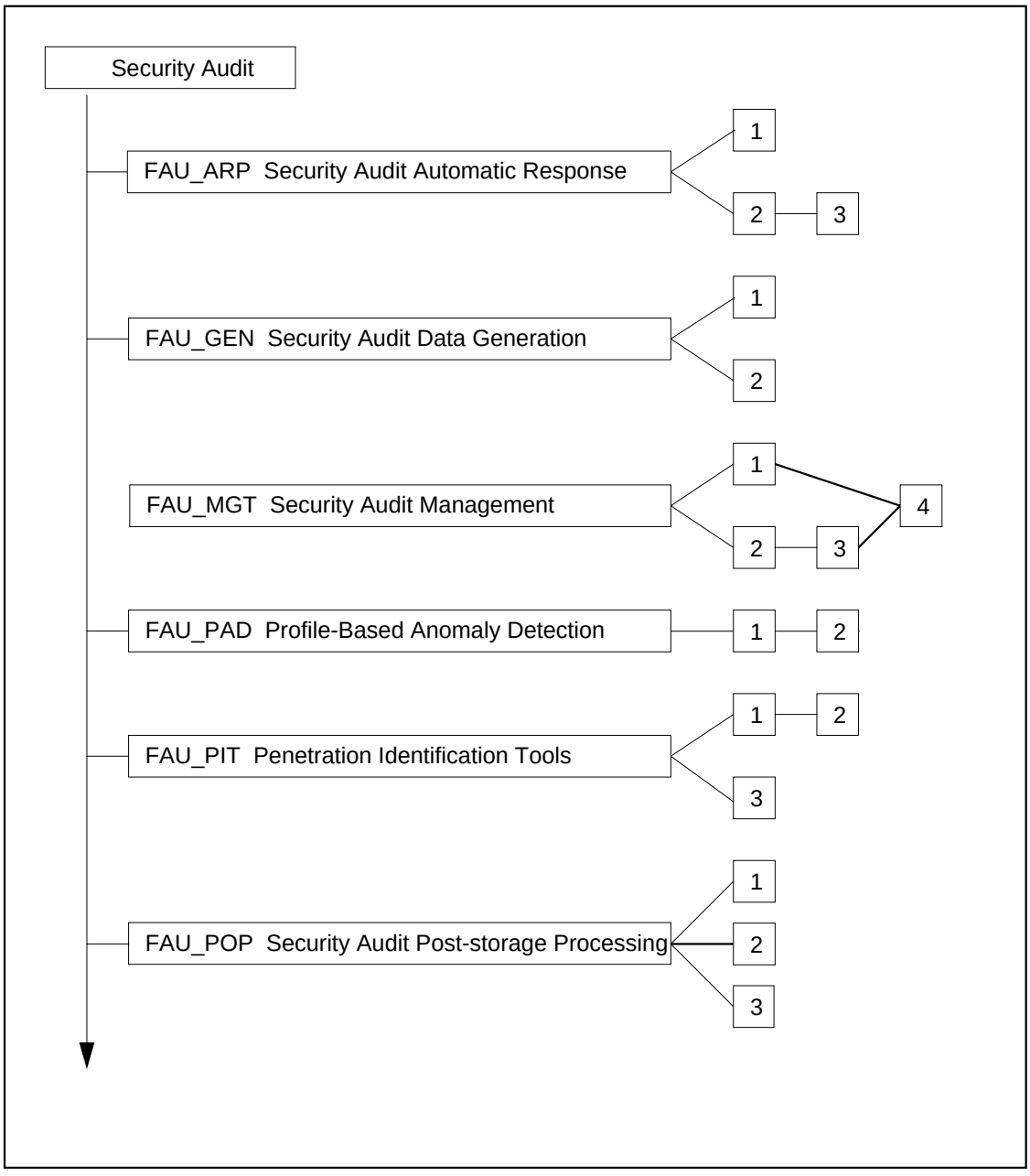

Figure 1.5 - Security Audit Class decomposition

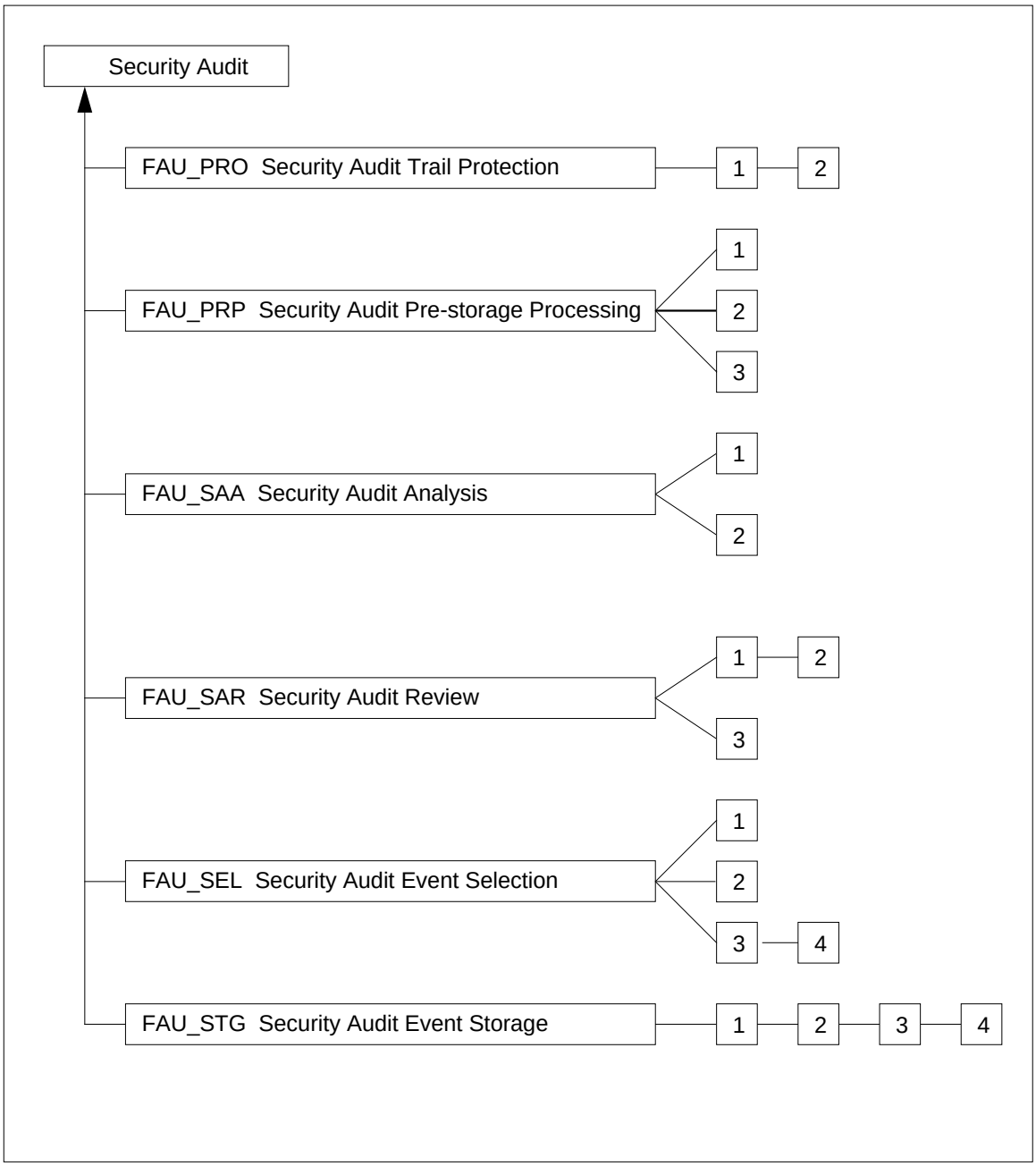

Figure 1.6 - Security Audit Class decomposition (Cont.)

FAU_ARP Security Audit Automatic Response

39 The Security Audit Automatic Response family describes requirements for the handling of audit events after the pre-storage processing and analysis functions (if any) were performed on the information. The possibilities include requirements for alarms or TSF action (automatic response). For example, the TSF could include the generation of real time alarms, termination of the offending process, disabling of a service, or disconnection or invalidation of a user account.

Application Notes

40 An audit event appears to be a condition under which the TSF reacts to an "imminent security violation" if:

a) it is part of the set of auditable events, defined in FAU_SAA Security Audit Analysis, whose occurrence or accumulated occurrence indicates a potential imminent violation of the TSP; and

b) the TSF has detected, using the relevant FAU_SAA Security Audit Analysis component, the occurrence or aggregate occurrence of the prior audit events which indicates that a potential violation is imminent.

41 The set of auditable events could be defined in the relevant component by the PP/ ST author for a specific period of time (e.g., date, duration).

Documentation notes

42 The following assurance documentation (if applicable) should contain the following information:

a) Recommendations for handling notifications generated by the TSF when a security violation appears imminent [AGD_ADM Administrator guidance];

b) Recommendations for ending the recurrence of security relevant events when a security violation appears imminent [AGD_ADM Administrator guidance];

c) Recommendation for managing "least disruptive action" to terminate aberrant behaviour [AGD_ADM Administrator guidance].

FAU_ARP.1 Security Alarms

User Application Notes

43 A warning should be presented to the authorised administrator for follow up action in the event of an alarm. The delay of delivery should be carefully considered by the PP/ST author.

FAU_ARP.2 Automatic Response

User Application Notes

44 Consideration should be given to the threat of using the automated responses to maliciously force unexpected availability failure (e.g., such automated responses may be exploited to force a system shutdown).

Operations

Assignment:

45 **For FAU_ARP.2.1, the PP/ST author should define [*the least disruptive actions*] to be automatically taken by the TSF to disregard or terminate the occurrence of offending events when a security violation appears imminent. For example, the TSF could specify termination of the offending processes or close the user session.**

FAU_ARP.3 Configurable Automatic Response

User Application Notes

46 Consideration should be given to the threat of using the automatic responses to maliciously force unexpected availability failure (e.g., force all external connections of the system down).

Evaluator application notes

47 Any of the least disruptive actions defined by the PP/ST author in the list should be supported by the TSF to terminate the occurrence of security relevant events.

Operations

Assignment:

48 **For FAU_ARP.3.1, the PP/ST author should define a [*list of the least disruptive actions*] from which the authorised administrator could select the actions to be automatically taken by the TSF to terminate the occurrence of security relevant events when a security violation appears imminent. For example, the PP/ST author could specify termination of the offending processes or close the user session and the authorised administrator chooses one or both.**

FAU_GEN Security Audit Data Generation

49 The Security Audit Data Generation family includes requirements to specify the audit events that should be generated by the TSF for some activity.

50 This family is presented in a manner which avoids a dependency on all components requiring audit support. Each component has an Audit section developed in which the events to be audited for that functional area are listed. When the PP/ST author assembles the PP/ST, the items in the audit area are used to complete the variable in this component. Thus, the specification of what could be audited for a functional area is localised in that functional area.

51 The list of auditable events is entirely dependent on the other functional families within the PP/ST. Each family definition should therefore include a list of its family-specific auditable events. Each auditable event in the list of auditable events specified in the functional family should correspond to one of the levels of audit event generation specified in this family (i.e., minimal, basic, detailed). This provides the PP/ST author with information necessary to ensure that all appropriate auditable events are specified in the PP/ST. The following example shows how auditable events are to be specified in appropriate functional families:

52 "The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful use of the user security attribute administration functions;

b) Basic: All attempted uses of the user security attribute administration functions;

c) Basic: Identification of which user security attributes have been modified; and

d) Detailed: With the exception of specific sensitive attribute data items (e.g., passwords, cryptographic keys), the new values of the attributes should be captured."

53 It should be observed that the categorisation of auditable events is hierarchical. For example, when Basic Audit Generation is desired, all auditable events identified as being both Minimal and Basic, should be included in the PP/ST through the use of the appropriate assignment operation, except when the higher level event simply provides more detail than the lower level event. When Detailed Audit Generation is desired, all identified auditable events (Minimal, Basic, and Detailed) should be included in the PP/ST.

Application Notes

54 The following are examples of the types of the events which should be defined as auditable within each PP/ST functional component:

a) Introduction of objects within the TSC into a subject's address space;

b) Deletion of objects;

c) Distribution or revocation of access rights or capabilities;

d) Changes to subject or object security attributes;

e) Policy checks performed by the TSF as a result of a request by a subject;

f) The use of access rights to bypass a policy check;

g) Use of Identification and Authentication functions;

h) Actions taken by an operator, and/or authorised administrator (e.g., suppression of a TSF protection mechanism as human-readable labels);

i) Import/export of data from/to removable media (e.g., printed output, tapes, diskettes).

Evaluator notes

55 It is difficult to list all auditable events during the PP/ST development. During the evaluation, additional auditable events may be discovered, and evaluator should check the list of auditable events for completeness.

Documentation notes

56 The following assurance documentation (if applicable) should contain the following information:

a) The list of auditable events [AGD_ADM Administrator guidance].

b) The audit record format [AGD_ADM Administrator guidance].

FAU_GEN.1 Audit Data Generation

User Application Notes

57 This component defines requirements to identify the auditable events for which audit records should be generated, and the information to be provided in the audit records. FAU_GEN.1 is for use when the TSP does not require that individual user identities be associated with audit events.

58 The information requested by this component to be recorded in each record is relevant for a general purpose operating system, but for some specific applications, a refinement of this information could be necessary to avoid requesting non available data.

Evaluator application notes

59 This component addresses the possible existence of audit functionality in the absence of individual user identities.

Operations

Selection:

60 **For FAU_GEN.1.1b, the PP/ST author should select the [*minimum, basic, detailed*] level of auditable events called out for audit in the Audit section of other functional components included in the PP/ST.**

Assignment:

61 **For FAU_GEN.1.1c, the PP/ST author should assign a list of [*other auditable events*] to be included in the list of auditable events, called out for minimal, basic, or detailed audit requirements in the Audit section of other functional components included in the PP/ST.**

Selection:

62 **For FAU_GEN.1.2a, the PP/ST author should select the [*success, failure*] of auditable events to be audited. This selection shall be consistent with the level of auditable events.**

Assignment:

63 **For FAU_GEN.1.2b, the PP/ST author should assign, for each auditable events included in the PP/ST, a list of [*other audit relevant information*] to be included in audit event records.**

FAU_GEN.2 User Identity Generation

User Application Notes

64 This component addresses the requirement in the TSP of accountability of auditable events at the level of individual user identity. This component should be used in addition to FAU_GEN.1 Audit Data Generation.

Evaluator application notes

65 This component addresses the capability for the TSF to connect any auditable event with individual user identities.

FAU_MGT Security Audit Management

66 The Security Audit Management family includes requirements relating to the management of the audit trail.

67 Examples of capabilities included in this family include:

- Creation, destruction, and emptying of audit trails;
- Control of the saturation of the audit trail; and
- Definition of warning points regarding the saturation of the audit trail.

Documentation notes

68 The following assurance documentation (if applicable) should contain the following information:

a) Recommendations for handling notifications generated by the TSF when audit trail saturation appears imminent [AGD_ADM Administrator guidance]; and

b) Recommendations for defining the limit to control the audit trail saturation [AGD_ADM Administrator guidance].

FAU_MGT.1 Audit Trail Management

User Application Notes

69 Consideration should be given to management operations that the authorised administrator is permitted to perform.

Operations

Selection:

70 **For FAU_MGT.1.1, the PP/ST author should select the set of operations that could be performed by the authorised administrator on the audit trail, from [*create, delete, empty*].**

FAU_MGT.2 Audit Trail Saturation Control

User Application Notes

71 Consideration should be given to the definition of the pre-defined limit of the size of the audit data in the audit trail, to provide the authorised administrator with the ability to take necessary actions after being warned in case of audit trail saturation.

Operations

Assignment:

72 **For FAU_MGT.2.1, the PP/ST author shall define the limit of the size of the audit data in the audit trail at which point an alarm will be generated. This limit should be expressed in percentage of the size of the audit trail, or in any other consistent manner.**

FAU_MGT.3 Audit Trail Saturation Management

User Application Notes

73 Consideration should be given by the authorised administrator when specifying the pre-defined limit of the size of the audit data in the audit trail, to provide the authorised administrator with the ability to take necessary actions after being alarmed in case of audit trail saturation.

FAU_MGT.4 Runtime Management

User Application Notes

74 This component addresses the requirement for the TSF to permit management of the TSF at the run time of the TOE. The management operations could include capability to delete, create, or empty the audit trail or define the pre-defined limit of the size of the audit data in the audit trail. Consideration should be given by the authorised administrator when managing the audit trail with conflicting with other audit operations such as storage, review or analysis.

FAU_PAD Profile-Based Anomaly Detection

75 The Profile-Based Anomaly Detection family primarily addresses the threat of unauthorised users who circumvent the authentication mechanisms of the system to exploit the privileges of an authorised user. This family defines requirements for an automated tool capable of constructing profiles that represent the historical patterns of usage for an individual user or group of users. The tool is then able to analyse new system usage against the profile and classify the new usage as either consistent or anomalous. A suspicion rating is calculated that represents how well new user activity corresponds to the established usage patterns represented in the user profile. When the suspicion rating exceeds a pre-defined threshold, the TSF indicates the potential violation to the FAU_ARP mechanism.

User notes

76 A *profile* is a structure that characterises the behaviour of users and/or subject; it represents how the users/subjects interact with the TSF in a variety of ways. Patterns of usage are established with respect to the various types of activity the users/subjects engage in (e.g., patterns in exceptions raised, patterns in resource utilisation (when, which, how), patterns in actions performed). The ways in which the various types of activity are recorded in the profile (e.g., resource measures, event counters, timers) are referred to as *profile metrics*.

77 Each profile represents the historical patterns of usage performed by members of the profile's *profile target group*. A profile target group refers to one or more users who interact with the TSF. The activity of each member of the profile group is used by the analysis tool in establishing the usage patterns represented in the profile. The following are some examples of profile target groups:

a) **Single user account**: one profile per user;

b) **Group ID or Group Account**: one profile for all users who possess the same group ID or operate using the same group account;

c) **Operating Role**: one profile for all users sharing a given operating role;

d) **System**: one profile for all users of a system.

78 Each member of a profile target group is assigned an individual *suspicion rating* that represents how closely that member's new activity corresponds to the established patterns of usage represented in the group profile.

79 The sophistication of the anomaly detection tool will largely be determined by the number of target profile groups required by the PP/ST and the complexity of the required profile metrics.

Documentation notes

80 The indicated assurance documentation (if applicable) should contain the indicated information:

a) administrative guidance for configuring the PAD functions.

FAU_PAD.1 Profile Based- Anomaly Detection

User Application Notes

81 The PP/ST author should specify the profile target group(s) analysed by the TSF.

82 The PP/ST author should define the profile metrics used to construct the usage profiles.

83 The PP/ST author should enumerate specifically what activity should be monitored by the TSF in order to perform its analysis. The PP/ST author should also identify specifically what information pertaining to the activity is necessary to construct the usage profiles.

84 FAU_PAD.1.1 requires that the TSF maintain profiles of system usage. The word maintain implies that the anomaly detector is actively updating the usage profile based on new activity performed by the profile target members. It is important here that the metrics for representing user activity are defined by the PP/ST author. For example, there may be a thousand different actions an individual may be capable of performing, but the anomaly detector may choose to monitor a subset of that activity. Anomalous activity gets integrated into the profile just like non-anomalous activity (assuming the tool is monitoring those actions). Things that may have appeared anomalous four months ago, might over time become the norm (and vice-versa) as the user's work duties change. The TSF wouldn't be able to capture this notion if it filtered out anomalous activity from the profile updating algorithms.

85 Administrative notification should be provided such that the administrator understands the significance of the suspicion rating.

86 The PP/ST author should define how to interpret suspicion ratings and the conditions under which anomalous activity is indicated to the FAU_ARP mechanism.

Operations

Assignment:

87 **For FAU_PAD.1.1, the PP/ST author should [*specify the profile target group*]. A single PP/ST may include multiple profile target groups.**

88 **For FAU_PAD.1.3, the PP/ST author should [*specify conditions under which anomalous activity is reported by the TSF*]. Conditions may include the suspicion rating reaching a certain value, or based on the type of anomalous activity observed.**

FAU_PAD.2 Dynamic Profile-Based Surveillance and Response

User Application Notes

89 The PP/ST author should specify the profile target group(s) analysed by the TSF.

90 The PP/ST author should define the profile metrics used to construct the usage profiles.

91 The PP/ST author should enumerate specifically what activity should be monitored by the TSF in order to perform its analysis. The PP/ST author should also identify specifically what information pertaining to the activity is necessary to construct the usage profiles.

92 Administrative notification should be provided such that the administrator understands the significance of the suspicion rating, and what possible responses might be appropriate given the rating.

93 The elements of FAU_PAD.3.4 require that the TSF implementing the runtime surveillance be able to perform actions to terminate activity from the subject(s) whose suspicion rating exceeds a pre-defined threshold. FAU_PAD.3.4 requires the analysis tool to operate during runtime (i.e., as the system activity is being performed), where it is able to perform an automated response.

94 The PP/ST author should define how to interpret suspicion ratings and the conditions under which anomalous activity is reported to the administrator.

Operations

Assignment:

95 **For FAU_PAD.2.1, the PP/ST author should [*specify the profile target group*]. A single PP/ST may include multiple profile target groups.**

96 **For FAU_PAD.2.3, the PP/ST author should [*specify conditions under which anomalous activity is reported by the TSF*]. Conditions may include the suspicion rating reaching a certain value, or based on the type of anomalous activity observed.**

97 **For FAU_PAD.2.4, the PP/ST author should specify the *list of profile metrics subject to dynamic configuration.***

98 **For FAU_PAD.2.5, the PP/ST author should identify the conditions under which *anomalous activity is reported to the administrator or action is taken to terminate further activity from the responsible individual.***

FAU_PIT Penetration Identification Tools

99 The Penetration Identification Tools family primarily addresses the threat of users who perform malicious activity on the TOE. The family defines requirements for the development of tools capable of analysing system activity against a known set of events whose occurrence or combination represents suspicious, if not illegal activity. Such events are worthy of immediate attention by a human reviewer, and may further require an automated response by the analysis tool.

User notes

100 In practice, it is at best rare when an analysis tool can detect with certainty when a security violation is imminent. However, there do exist some system events that are so significant they are always worthy of independent review. Example of such events include the deletion of a key TSF security data file (e.g., the password file) or activity such as a remote user attempting to gain administrative privilege. These events are referred to as *signature events* in that their occurrence in isolation from the rest of the system activity are indicative of intrusive activity.

101 The levelling of this family is intended to recognise high-level differences in the functional capabilities of Penetration Identification Tools. The distinctions considered significant in this family are:

1) complexity of heuristics (signature event analysis vs. multi-step intrusion scenarios);

2) runtime analysis vs. batch-mode post-collection analysis; and

3) dynamic tool configuration vs. static configuration.

102 The complexity of a given tool will depend greatly on the assignments defined by the PP/ST author in identifying the base set of signature events and event sequences.

103 The PP/ST author should define a base set of signature events to be represented by the TSF. Additional signature events may be defined by the system developer.

104 The PP/ST author should enumerate specifically what events should be monitored by the TSF in order to perform the analysis. The PP/ST author should identify specifically what information pertaining to the event is necessary to determine if the event maps to a signature event.

105 Administrative notification should be provided such that the administrator understands the significance of the event and what possible responses might be appropriate.

106 An effort was made in the specification of these requirements to avoid a dependency on audit data as the sole input for monitoring system activity. This was done in recognition of the existence of previously developed intrusion detection tools that do not perform their analyses of system activity solely through the use of audit data (examples of other input data include network datagrams, resource/ accounting data, or combinations of various system data). Levelling, therefore,

requires the PP/ST author to specify the type of input data used to monitor system activity.

Documentation notes

107 The indicated assurance documentation (if applicable) should contain the indicated information:

a) Identification of the signature events and event sequences represented in the tool and the significance of their appearance should they occur on the system; and

b) An explanation of how the administrator should respond to the occurrence of various signature events and event sequences.

FAU_PIT.1 Immediate Attack Heuristics

User Application Notes

108 The elements of FAU_PIT.1 do not require that the TSF implementing the immediate attack heuristics be the same TSF whose activity is being monitored. Thus, one can develop an intrusion detection component that operates independently of the system whose system activity is being analysed.

Operations

Assignment:

109 **For FAU_PIT.1.1, the PP/ST author should identify a base [*subset of system events*] whose occurrence, in isolation from all other system activity, may indicate a violation of the TSP. These include events that by themselves indicate a clear violation to the TSP, or whose occurrence is so significant they warrant human review.**

110 **In FAU_PIT.1.2, the PP/ST author should [*specify the information used to determine system activity*]. This information is the input data used by the analysis tool to determine the system activity that has occurred on the TOE. This data may include audit data, combinations of audit data with other system data, or may consist of data other than the audit data. The PP/ST author should define precisely what system events and event attributes are being monitored within the input data.**

FAU_PIT.2 Complex Attack Heuristics

User Application Notes

111 The PP/ST author should define a base set of penetration event sequences to be represented by the TSF. Additional penetration event sequences may be defined by the system developer.

112 The elements of FAU_PIT.2 do not require that the TSF implementing the complex attack heuristics be the same TSF whose activity is being monitored. Thus, one can develop an intrusion detection component that operates independently of the system whose system activity is being analysed.

Operations

Assignment:

113 **For FAU_PIT.2.1, the PP/ST author should identify a base set of [*ordered sequences of system events whose occurrence are representative of known penetration scenarios*]. These event sequences represent known multi-step penetration scenarios. Each event represented in the sequence should map to a monitored system event, such that as the system events are performed, they are bound (mapped) to the known penetration event sequences.**

114 For FAU_PIT.2.1, the PP/ST author should identify a base [*subset of system events*] whose occurrence, in isolation from all other system activity, may indicate a violation of the TSP. These include events that by themselves indicate a clear violation to the TSP, or whose occurrence is so significant they warrant human review.

115 In FAU_PIT.2.2, the PP/ST author should [*specify the information used to determine system activity*]. This information is the input data used by the analysis tool to determine the system activity that has occurred on the TOE. This data may include audit data, combinations of audit data with other system data, or may consist of data other than the audit data. The PP/ST author should define precisely what system events and event attributes are being monitored within the input data.

FAU_PIT.3 Dynamic Run-Time Surveillance and Management

User Application Notes

116 The PP/ST author should define a base set of penetration event sequences to be represented by the TSF. Additional penetration event sequences may be defined by the system developer.

117 For each signature event and penetration event sequence, the PP/ST author should specify the action(s) to be taken by the TSF.

118 The elements of FAU_PIT.3 require that the TSF implementing the runtime surveillance be able to perform actions to terminate activity from the subject(s) responsible for the signature event or penetration event sequence. FAU_PIT.4.4 requires the analysis tool to operate during runtime (i.e., as the system activity is being performed), where it is able to perform an automated response to subjects whose activity match known intrusive activity.

Evaluator application notes

119 Developing an intrusion detection tool truly able to perform a real-time analysis is somewhat difficult. For example, an audit trail analysis tool may suffer a delay between the time an action is performed by a process and the time the audit record for that action reaches the tool for analysis. A runtime analysis tool is distinguished from a post-storage analysis tool in that a runtime tool will perform its analysis on an audit record the moment the audit record is made available by the audit generation mechanism. Internal buffering of audit records by the audit generation mechanism may be found to pose an unacceptable time delay.

Operations

Assignment:

120 **For FAU_PIT.3.1, the PP/ST author should specify whether the authorised administrator shall be able to [*add, modify, and/or delete*] signature events and event sequences from the TSF's knowledge base.**

FAU_POP Security Audit Post-storage Processing

121 The Security Audit Post-storage Processing family presents requirements for taking stored audit data, for which the TSF could have employed techniques to reduce the volume of data collected, and processing it into a useful form. The result could be useful for a human user (e.g., human-readable presentation), for a process (e.g., compressed format for transfer), or for machine user (e.g., machine independent representation).

Application Notes

122 If the audit data stored in an audit trail need to be transferred or stored for an extended period of time, identifying users by internal process identifiers or files by internal file identifiers would be of little value. Those internal identifiers would only be of value on the host machine where the records were generated, and only for a limited period of time, until they were recycled.

FAU_POP.1 Human Understandable Format

User Application Notes

123 This component is used to transform the audit data stored in the audit trail in a form understandable by human users. The informational content of the audit trail is not modified by this transformation.

124 For example, this transformation could be performed on a User identifier (e.g., Unix UID) to generate a symbolic user name, or on file identifier (e.g., Unix Inode) to generate a symbolic name in the file structure.

FAU_POP.2 Automated Treatment Format

User Application Notes

125 This component is used to transform the audit data stored in the audit trail into a format independent from the host machine or system and consistent with the target process or TSF. The informational content of the audit trail is not modified by this transformation.

126 For example, audit data could be transformed to support different representations of floating numbers, or audit data could be compressed before transfer.

FAU_POP.3 Flexible Format

User Application Notes

127 This component is used to provide the ability to restructure and reorder information provided in the audit trail before export or use. The informational content of the

audit trail is not modified by this transformation, a new representation of the audit data could be generated.

128 Rearranging the content of the audit data might result in information from separate audit records being grouped together to generate a single audit record in a more useful audit trail format.

129 Rearranging the order of audit data might also result in information generated in a distributed environment and stored together being reordered to provide the events in a time ordered presentation.

FAU_PRO Security Audit Trail Protection

130 The Security Audit Trail Protection family defines requirements to protect information stored in the audit trail or results of audit data processing against unauthorised disclosure or modification.

Application Notes

131 The level of protection of this information should be consistent with the level of protection of the TOE information.

132 Modification or deletion of this information should be limited to the authorised administrators. Direct read access is also limited to the authorised administrators, but authorised users should grant access using relevant review, analysis or processing tools.

133 The protection of audit data and / or audit analysis results could be enforced in a distributed environment between two or more TSF by the use of FTP_ITC Inter-TSF Trusted Channel components.

Documentation notes

134 The following assurance documentation (if applicable) should contain the following information:

a) Description of the protection rules for the audit trail. [AGD_ADM Administrator guidance].

b) The rules for managing access to audit trail by users. [AGD_ADM Administrator guidance].

FAU_PRO.1 Restricted Audit Trail Access

User Application Notes

135 At this level, only the authorised administrator may access the audit trail, to read, modify, or delete information.

FAU_PRO.2 Extended Audit Trail Access

User Application Notes

136 At this level, only the authorised administrator has full access to the audit trail, to read, modify, or delete information. Other authorised users may be granted limited read access to the audit trail, in accordance with the TSP.

Operations

Assignment:

137 **In FAU_PRO.2.2, the PP/ST author should assign the [*list of audit information*] the authorised users could read.**

FAU_PRP Security Audit Pre-storage Processing

138 The Security Audit Pre-storage Processing family describes requirements for ensuring that the audit information is provided in a consistent format for its subsequent attempted use (e.g., transfer, storage, review or analysis).

Application Notes

139 For time optimisation or data size constraints, the audit data generated could be provided in a useful format for automatic treatment or storage, but not understandable by a human user. Also, information might need to be compressed before storage or transfer.

140 Some specific attention should be given to the network and distributed environment regarding audit data processing. The appropriate audit requirements for networks and other large systems differ significantly from those for a stand-alone system. Different hosts and servers in a distributed environment could have differing naming policies and values. The transfer and use of this information could require the definition of an agreed format.

FAU_PRP.1 Human Understandable Format

User Application Notes

141 This component should be used prior to review or analysis, to provide a human understandable presentation of any audit data generated for an event (e.g., replace UID by user name, or Inode number by file pathname in a unix environment).

Evaluator application notes

142 If the TSF does not provide an association between a symbolic name and an object, this requirement does not apply. This association need only be correct as of the time the event was recorded.

FAU_PRP.2 Automated Treatment Format

User Application Notes

143 This component should be requested prior to transfer or analysis of the audit data is performed, to provide information in a useful format for automated treatment by the TSF itself or any other machine user. If the format of the information is modified, the reverse operation should be provided to be able to use this information.

FAU_PRP.3 Flexible Format

User Application Notes

144 This component could be requested to store audit data in a coherent audit trail if those data were generated by different part of the TOE with differing formats.

Evaluator application notes

145 Consideration should be given to the rearranging option, to avoid arbitrary results from being generated.

FAU_SAA Security Audit Analysis

146 The Security Audit Analysis family defines requirements related to intelligent tools that perform analysis of the audit information, prior to it being placed in the audit trail (to detect possible imminent violation) or after storage (to detect potential violation).

147 The action to be performed by the TSF on detection of a possible imminent or potential violation is defined in FAU_ARP Security Audit Automatic Response components.

Application Notes

148 For real-time analysis, audit data could be transformed into a useful format for automated treatment, but into a different useful format for delivery to authorised users for review.

Documentation notes

149 The following assurance documentation (if applicable) should contain the following information:

a) Description of the effect of accumulation or combination of events [AGD_ADM Administrator guidance]; and

b) The rules for managing the violation analysis events set and rules [AGD_ADM Administrator guidance].

FAU_SAA.1 Imminent Violation Analysis

User Application Notes

150 This component is used to specify the set of auditable events whose occurrence or accumulated occurrence held to indicate a potential violation of the TSP, and any rules to be used to perform the violation analysis.

Operations

Assignment:

151 **For FAU_SAA.1.2.a, the PP/ST author should identify the [*subset of defined auditable events*] whose occurrence or accumulated occurrence need to be detected as an indication of a potential violation of the TSP.**

Assignment:

152 **In FAU_SAA.1.2.b, the PP/ST author should assign [*any other rules*] which the TSF shall use in its analysis of the audit trail. Those rules could include specific requirements to express the need for the events to occur in a certain period of time (e.g., period of the day, duration).**

FAU_SAA.2 Configurable Violation Analysis

User Application Notes

153 This component is used to specify the set of auditable events whose occurrence or accumulated occurrence indicates a potential violation of the TSP, and any rules to be used to perform the violation analysis. This set of events or rules could be modified by the authorised administrator, through addition, modification or deletion of events or rules.

Operations

Selection:

154 **In FAU_SAA.2.1, the PP/ST author should select from [*addition, modification, deletion*] the permitted operations the authorised administrator could perform on the set of rules used for the analysis.**

FAU_SAR Security Audit Review

155 The Security Audit Review family defines requirements related to intelligent tools that perform review of the audit information.

156 These tools should allow pre-storage or post-storage audit selection that includes, for example, the ability to selectively review:

- the actions of one or more users (e.g., identification, authentication, TOE entry, and access control actions);
- the actions performed on a specific object or TOE resource;
- all of a specified set of audited exceptions; or
- actions associated with a specific TSP attribute.

Application Notes

157 The distinction between review and selectable review is based on functionality. Review capability encompasses only the ability to view audit data. Selectable capability is more sophisticated, and requires the ability to perform searches based on a single criterion or multiple criteria with logical (i.e. and / or) relations, sort audit data, filter audit data, before reviewing audit data.

FAU_SAR.1 Restricted Audit Review

User Application Notes

158 This component is used to specify the need for audit review tools to provide authorised administrator with the ability to view all audit data.

FAU_SAR.2 Extended Audit Review

User Application Notes

159 This component is used to specify the need for audit review tools to provide authorised administrator with the ability to view all audit data and authorised users to view relevant audit data. The PP/ST author should refine the definition of the authorised users and the specification of the limitation of use of the tools in conformance with the TSP

FAU_SAR.3 Selectable Audit Review

User Application Notes

160 This component is used to specify that audit review tools should perform selection of the audit data to be reviewed. If based on a single criterion, this component could be used more than one time, to define different single criteria that could be used to perform the analysis. If based on multiple criteria, those criteria should be related

together with logical (i.e. and / or) relations, and the tools should provide the ability to manipulate audit data (e.g., sort, filter).

Operations

Assignment:

161 **For FAU_SAR.3.1, the PP/ST author should assign [*multiple criteria with logical relations*] to be used to select the audit data for review.**

FAU_SEL Security Audit Event Selection

162 The Security Audit Event Selection family provides requirements related to the capabilities of identifying which of the possible auditable events are to be audited. The auditable events are defined in the FAU_GEN Security Audit Data Generation family, but those events should be defined as being selectable in this component to be audited.

Application Notes

163 This family ensures that it is possible to keep the audit trail from becoming so large that it becomes useless, by defining the appropriate granularity of the selected security audit events.

Documentation notes

164 The following assurance documentation (if applicable) should contain the following information:

a) Description of the selection rules for the audit events. [AGD_ADM Administrator guidance]; and

b) The rules for managing the auditable set of events. [AGD_ADM Administrator guidance].

FAU_SEL.1 Selective Audit

User Application Notes

165 This component defines the criteria used for the selection of events to be audited. Those criteria could permit inclusion or exclusion of events from the set of auditable events, based on user attributes, subject attributes, objects attributes, or event types.

166 The existence of individual user identities is not assumed for this component. This would allow for TOEs such as routers that may not support the notion of users.

167 For a distributed environment, the Host identity could be used as a selection criteria for events to be audited.

Operations

Selection:

168 **For FAU_SEL.1.1a, the PP/ST author should select from [*Object identity, User identity, Subject identity, Host identity, Event Type*] the security attributes that audit selectivity is based upon.**

Assignment:

169 **For FAU_SEL.1.1b, the PP/ST author should specify the [*list of additional attributes*] that audit selectivity is based upon.**

FAU_SEL.2 Runtime Selection Mode

Evaluator application notes

170 The function to select or change the set of audited events should be available during TSF operation. The usage of this capability should be restricted to the authorised administrator.

FAU_SEL.3 Restricted Runtime Display Mode

Evaluator application notes

171 The TSF should provide to the authorised administrator a capability to display the currently selected events.

FAU_SEL.4 Extended Runtime Display Mode

Evaluator application notes

172 The TSF should provide a capability to display the currently selected events. The full usage of this capability should be restricted to the authorised administrator although limited use may be provided to authorised users.

FAU_STG Security Audit Event Storage

173 The Security Audit Event Storage family describes requirements for storing audit data for later use, including requirements controlling the loss of audit information due to system failure, attack and/or exhaustion of storage space.

Application Notes

174 The permanence of the audit trail should be considered also in terms of duration of validity of the audit information.

Documentation notes

175 The following assurance documentation (if applicable) should contain the following information:

a) Conditions under which loss of audit data due to system failure should be enumerated and the potential number of audit events lost should be documented [AGD_ADM Administrator guidance].

FAU_STG.1 Permanent Audit Trail Storage

User Application Notes

176 In a distributed environment, as the location of the audit trail should be in the TSC, but not necessarily co-located with the function generating the audit data, the PP/ ST author could request authentication of the originator of the audit record, or non repudiation of the origin of the record prior storing this record in the audit trail.

FAU_STG.2 Enumeration of Audit Data Loss

Evaluator application notes

177 No functional requirements are required by this component, and it could be satisfied by provision of the relevant information in the documentation.

Operations

Selection:

178 **In FAU_STG.2.2, the PP/ST author should select the condition [*audit storage exhaustion, failure, attack*] under which the TSF shall control audit data loss.**

FAU_STG.3 Prevention of Audit Data Loss

User Application Notes

179 This component requires that actions taken by the authorised administrator always be recorded in case of audit storage exhaustion. The PP/ST author should define whether the TSF should ignore or prevent the occurrence of auditable events to satisfy this requirement. Consideration should be given to the choice of the action to be taken by the TSF in the case of audit storage exhaustion, as ignoring events, which provides better availability of the TOE, will also permit actions to be performed without being recorded and without the user being accountable.

Operations

Selection:

180 **In FAU_STG.3.2, the PP/ST author should select the condition [*audit storage exhaustion, failure, attack*] under which the TSF shall control audit data loss.**

181 **In FAU_STG.3.3, the PP/ST author should select the action [*ignoring, preventing*] the occurrence of auditable actions, by which the TSF shall control audit data loss.**

FAU_STG.4 Manageable Prevention of Audit Data Loss

User Application Notes

182 This component requires that actions taken by the authorised administrator always be recorded in case of audit storage exhaustion. The PP/ST author should define whether the TSF should ignore or prevent the occurrence of auditable events to satisfy this requirement. Consideration should be given to the choice of the action to be taken by the TSF in the case of audit storage exhaustion, as ignoring events, which provides better availability of the TOE, will also permit actions to be performed without being recorded and without the user being accountable.

Operations

Selection:

183 **In FAU_STG.4.2, the PP/ST author should select the condition [au*dit storage exhaustion, failure, attack*] under which the TSF shall control audit data loss.**

Class FCO

Communication

184 This class describes requirements specifically of interest for TOEs which are used for the transport of information. The currently identified families deal with non-repudiation.

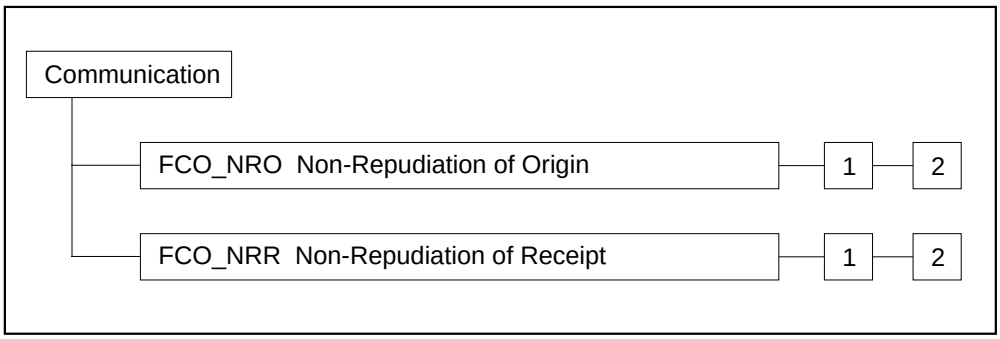

Figure 1.7 - Communication class decomposition

185 Figure 1.7 shows the decomposition of this class into its constituent components.

186 In this class the concept of “information” is being used. This information should be interpreted as the object being communicated. Therefore this information could contain an electronic mail message, a file, or a set of predefined attribute types.

187 In the literature the terms ‘proof of receipt’ and ‘proof of origin’ are commonly used terms. However it is recognised that the term ‘proof’ might be interpreted in a legal sense. To avoid this problem the components use the word ‘evidence’ instead of ‘proof’.

FCO_NRO Non-Repudiation of Origin

188 Non-repudiation of origin defines requirements to provide evidence to users/ subjects about, for example, the identity of the originator of some information. The originator cannot successfully deny sending the information because evidence of origin (e.g. digital signature) provides evidence of the binding between the originator and the information sent. The recipient or a third party can verify the evidence of origin.

User notes

189 The Non-repudiation of Origin requirements provide evidence to other subjects about the attributes of the originator of information. This evidence should not be forgeable.

190 If a part of the protected part of the information or of the associated attributes is altered in any way, validation of the evidence of origin may fail. Therefore a PP/ST author might consider including integrity mechanisms such as FDP_UIT.1 Data Exchange Integrity in the PP/ST.

191 In non-repudiation there are several different roles involved, each of which could be combined in one or more subjects. The first role is a subject that requests evidence of origin (only in FCO_NRO.2 Selective Proof of Origin). The second role is the recipient and/or other subjects to which the evidence is provided, (e.g. a notary). The third role is a subject that requests verification of the evidence of origin, for example a recipient or a third party like an arbiter.

192 The PP/ST must specify the conditions which must be met to be able to verify the validity of the evidence. Such a condition could be a time interval, or related to reserved memory, or the availability of third parties. These conditions therefore allow the tailoring of the non-repudiation to national requirements such as being able to provide evidence for several years.

193 In most cases, the identity of the recipient will be the identity of the user who received the transmission. In some instances, the PP/ST author does not want the user identity to be exported. In that case the PP/ST author must consider whether he wants to include this class, or whether the identity of the transport service provider, or the identity of the host should be used.

194 In addition to, or instead of, the user identity a PP/ST author might be more concerned about the time the information was transmitted. For example, requests for proposals must be transmitted before a certain date in order to be considered. The requirements can, in such instances, be customised to provide a timestamp indication (time of origin).

FCO_NRO.1 Enforced Proof of Origin

Operations

Assignment:

195 **In FCO_NRO.1.1 the PP/ST author should fill in the types of *information* subject to the Proof of Origin function, for example electronic mail messages.**

196 **In FCO_NRO.1.2 the PP/ST author should fill in the *list of attributes* with the attributes which shall be linked to the information, for example originator identity, time of origin, and location of origin.**

197 **In FCO_NRO.1.2 the PP/ST author should fill in the *list of information fields* within the information over which the attributes provide evidence of origin, such as the body of the information.**

198 **In FCO_NRO.1.3 the PP/ST author should fill in the *list of limitations* under which the evidence can be verified. For example the evidence can only be verified within a 24 hour time interval. An assignment of 'immediate' or 'indefinite' is acceptable.**

Selection:

199 **In FCO_NRO.1.3 the PP/ST author should specify the user/subject who can verify the evidence of origin.**

Assignment:

200 **In FCO_NRO.1.3 the PP/ST author, dependent on the selection, should specify the *third parties* that can verify the evidence of origin.**

FCO_NRO.2 Selective Proof of Origin

Operations

Assignment:

201 **In FCO_NRO.2.1 the PP/ST author should fill in the types of *information* subject to the Proof of Origin function, for example electronic mail messages.**

202 **In FCO_NRO.2.2 the PP/ST author should fill in the *list of attributes* with the attributes which shall be linked to the information, for example originator identity, time of origin, and location of origin.**

203 **In FCO_NRO.2.2 the PP/ST author should fill in the *information fields* within the information over which the attributes provide evidence of origin, such as the body of the information.**

204 **In FCO_NRO.2.3 the PP/ST author should fill in the *list of limitations* under which the evidence can be verified. For example the evidence can only be verified within a 24 hour time interval. An assignment of 'immediate' or 'indefinite' is acceptable.**

Selection:

205 **In FCO_NRO.2.3 the PP/ST author should specify the user/subject who can verify the evidence of origin.**

Assignment:

206 **In FCO_NRO.2.3 the PP/ST author, dependent on the selection, should specify the *third parties* that verify the evidence.**

Selection:

207 **In FCO_NRO.2.4 the PP/ST author should specify the user/subject that can request evidence of origin.**

Assignment:

208 **In FCO_NRO.2.4 the PP/ST author, if applicable, should specify the *third parties* that can request evidence of origin.**

FCO_NRR Non-Repudiation of Receipt

209 Non-repudiation of receipt defines requirements to provide evidence to other users/ subjects about, for example, that the information was received by the recipient. The recipient cannot successfully deny receiving the information because evidence of receipt (e.g. digital signature) provides evidence of the binding between the recipient attributes and the information. The originator or a third party can verify the evidence of receipt.

User notes

210 The Non-repudiation of Receipt requirements provide a requirement to provide evidence to other subjects about the attributes of the recipient of the information. This evidence should not be forgeable.

211 If the information or the associated attributes are altered in any way, validation of the evidence of receipt with respect to the original information might fail. Therefore a PP/ST author might consider including integrity mechanisms such as FDP_UIT.1 Data Exchange Integrity in the PP/ST.

212 In non-repudiation there are several different roles involved, each of which could be combined in one or more subjects. The first role is a subject that requests evidence of receipt (only in FCO_NRR.2 Selective Proof of Receipt). The second role is the recipient and/or other subjects to which the evidence is provided, (e.g., a notary). The third role is a subject that requests verification of the evidence of origin, for example a recipient or a third party like an arbiter.

213 The PP/ST author must specify the conditions which must be met to be able to verify the validity of the evidence. Such a condition could be a time interval, related to reserved memory, or the availability of third parties. These conditions therefore allow the tailoring of the non-repudiation to national requirements such as being able to provide evidence for several years.

214 In most cases, the identity of the recipient will be the identity of the user who received the transmission. In some instances, the PP/ST author does not want the user identity to be exported. In that case the PP/ST author must consider whether he wants to include this class, or whether the identity of the transport service provider, or the identity of the host should be used.

215 In addition to, or instead of, the user identity a PP/ST author might be more concerned about the time the information was received. For example, when an offer expires at a certain date, orders must be received before a certain date in order to be considered. The requirements can, in such instances, be customised to provide a timestamp indication (time of receipt).

FCO_NRR.1 Enforced Proof of Receipt

Operations

Assignment:

216 **In FCO_NRR.1.1 the PP/ST author should fill in the types of *information* subject to the Proof of Receipt function, for example electronic mail messages.**

217 **In FCO_NRR.1.2 the PP/ST author should specify the *list of attributes* which shall be linked to the information, for example recipient identity, time of receipt, and location of receipt.**

218 **In FCO_NRR.1.2 the PP/ST author should specify the *list of information fields* with the fields within the information over which the attributes provide evidence of receipt, such as the body of the information.**

219 **In FCO_NRR.1.3 the PP/ST author should specify the *list of limitations* under which the evidence can be verified. For example the evidence can only be verified within a 24 hour time interval. An assignment of 'immediate' or 'indefinite' is acceptable.**

Selection:

220 **In FCO_NRR.1.3 the PP/ST author should specify the user/subjects who can verify the evidence of receipt.**

Assignment:

221 **In FCO_NRR.1.3 the PP/ST author, dependent on the selection, should specify the *third parties* that can verify the evidence of receipt.**

FCO_NRR.2 Selective Proof of Receipt

Operations

Assignment:

222 **In FCO_NRR.2.1 the PP/ST author should fill in the types of *information* subject to the Proof of Receipt function, for example electronic mail messages.**

223 **In FCO_NRR.2.2 the PP/ST author should specify the *list of attributes* which shall be linked to the information, for example recipient identity, time of receipt, and location of receipt.**

224 **In FCO_NRR.2.2 the PP/ST author should specify the *list of information fields* with the fields within the information over which the**

attributes provide evidence of receipt, such as the body of the information.

225 **In FCO_NRR.2.3 the PP/ST author should specify the *list of limitations* under which the evidence can be verified. For example the evidence can only be verified within a 24 hour time interval. An assignment of 'immediate' or 'indefinite' is acceptable.**

Selection:

226 **In FCO_NRR.2.3 the PP/ST author should specify the user/subjects who can verify the evidence of receipt.**

Assignment:

227 **In FCO_NRR.2.3 the PP/ST author, dependent on the selection, should specify the *third parties* that can verify the evidence of receipt.**

Selection:

228 **In FCO_NRR.2.4 the PP/ST author should specify the user/subject who can request evidence of receipt.**

Assignment:

229 **In FCO_NRO.2.4 the PP/ST author, if applicable, should specify the *third parties* that can request evidence of origin.**

Class FDP

User Data Protection

230 This class contains families specifying requirements for TOE security functions and TOE security function policies related to protecting user data. This class differs from FIA and FPT in that FDP specifies components to protect user data, FIA specifies components to protect attributes associated with the user, and FPT specifies components to protect TSF information.

231 The class does not contain explicit requirements for TCSEC Mandatory Access Controls or Discretionary Access Controls; however, such requirements may be constructed using components from this class.

232 FDP does not explicitly deal with Confidentiality, Integrity, or Availability, as all three are most often intertwined in the policy and mechanisms. However, the TOE security policy must adequately cover these three policies in the PP/ST.

233 A final aspect of this class is that it specifies access control in terms of “operations”. An operation is defined as a specific type of access on a specific object. It depends on the level of abstraction of the PP/ST author whether these operations are described as “read” and/or “write” operations, or as more complex operations such as “update the database”.

234 The access control policy is concerned with the operations on the object. Information flow policies are concerned with the content of the object. Therefore, information flow policies are considered more in terms of flow of the information rather than a specific operation on an object.

235 This class is not meant to be a complete taxonomy of IT access policies, as others can be imagined. Those policies included here are simply those for which current experience with actual systems provides a basis for specifying requirements. There may be other forms of intent which are not captured in the definitions here.

236 For example, one could imagine a goal of having user-imposed (and user-defined) controls on information flow (e.g., an automated implementation of the NO FOREIGN handling caveat). However, this concept is not supported by existing practice, and research to date has not demonstrated practical general-purpose solutions, particularly in the context of a TOE supporting subjects that are not trusted to enforce that policy. Such concepts could, of course, be the subject of extensions to the FDP components.

237 Finally, it is important when looking at the components in FDP to remember that these components are requirements for functions which may be implemented by a mechanism which also serves or could serve another purpose. For example, it is possible to build an access control policy (FDP_ACC) which uses labels (FDP_IFF.1) as the basis of the access control mechanism.

238 A TOE security policy may encompass many security function policies (SFPs), each to be identified by the two policy oriented components FDP_ACC, and

FDP_IFC. These policies will typically take confidentiality, integrity, and availability aspects into consideration as required, to satisfy the TOE requirements. Care must be taken to ensure that all objects are covered by at least one SFP and that there are no conflicts arising from implementing the multiple SFPs.

239 Figures 1.8 and 1.9 show the decomposition of this class into its constituent components.

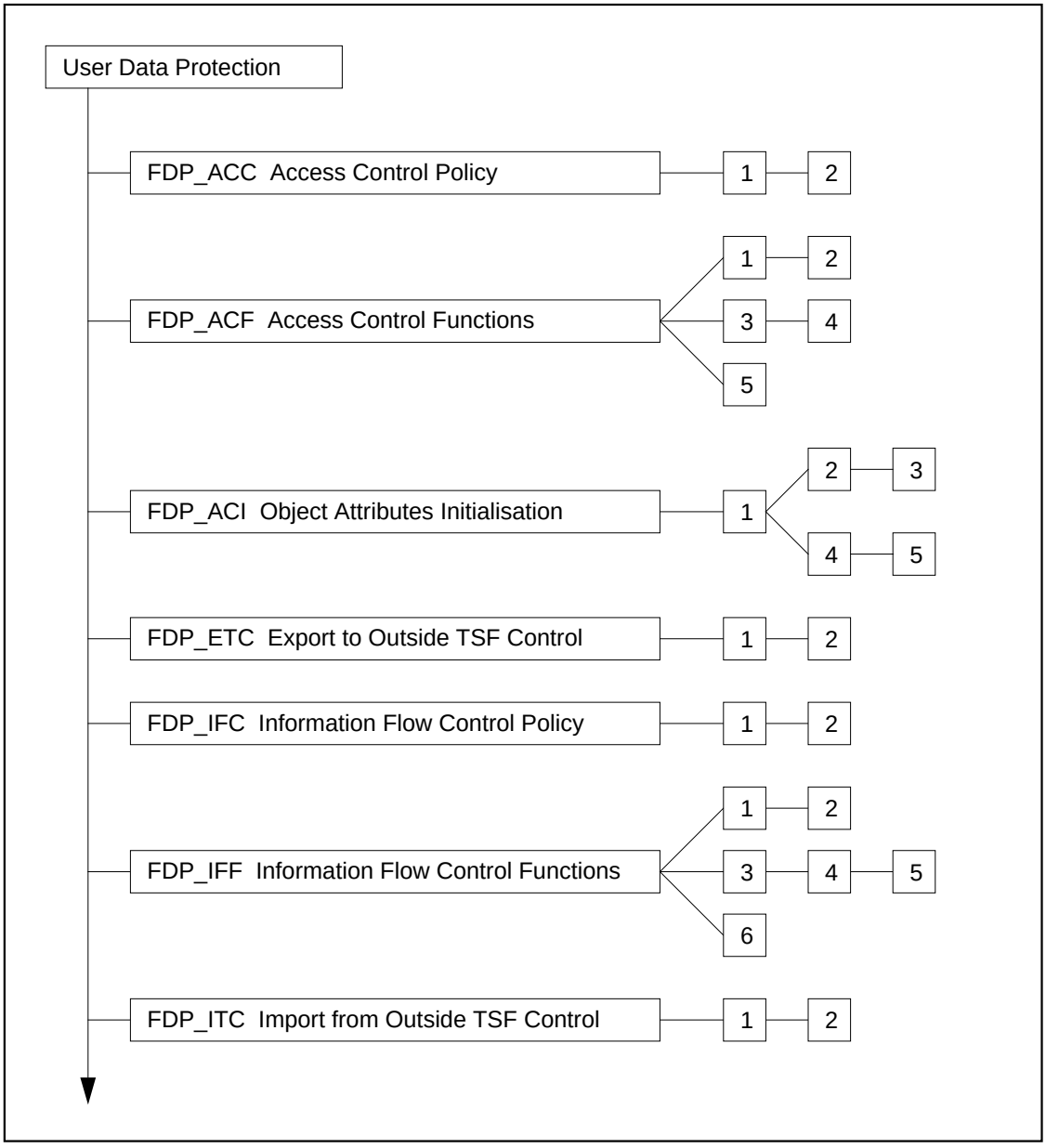

Figure 1.8 - User Data Protection class decomposition

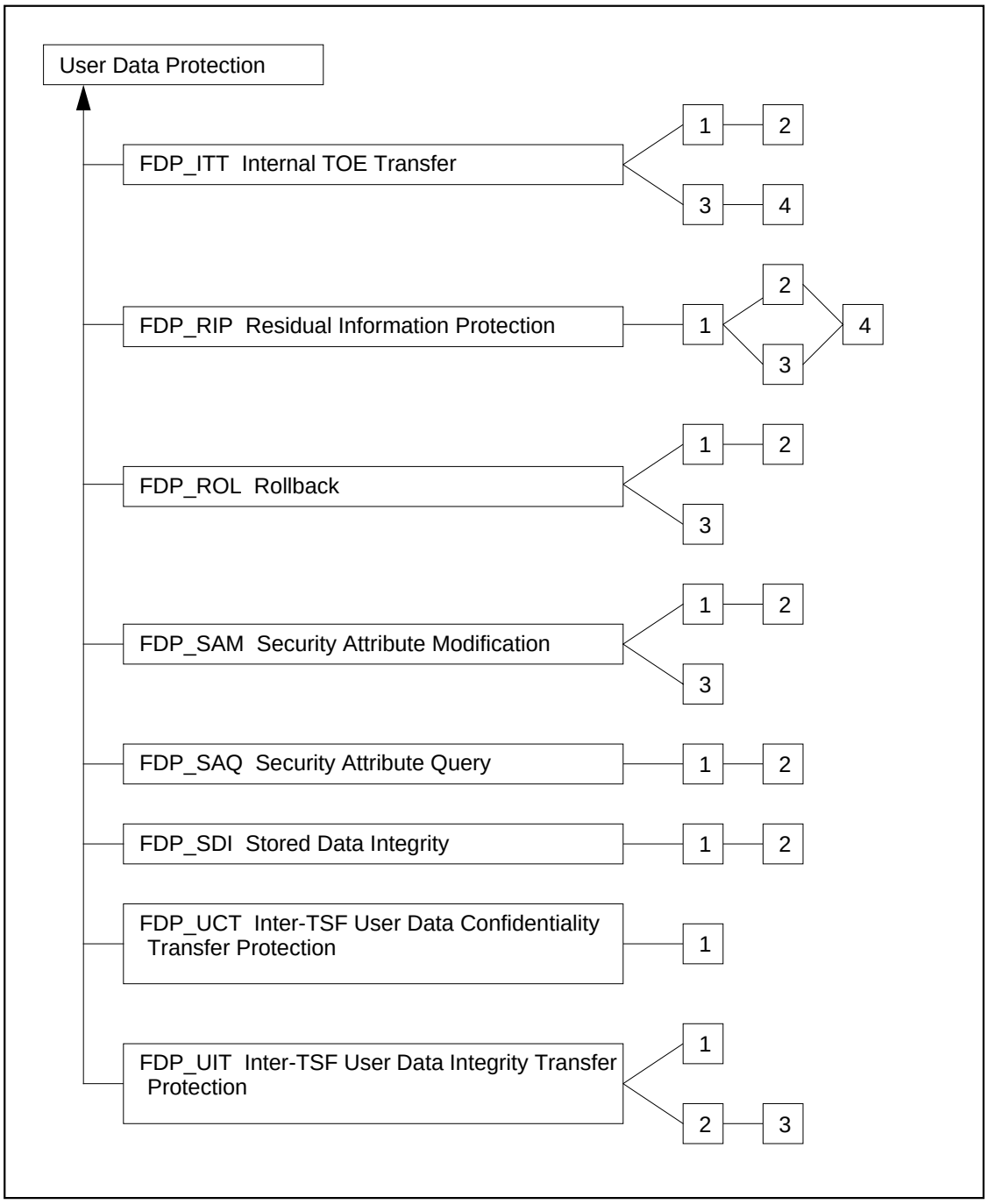

Figure 1.9 - User Data Protection class decomposition (cont.)

Construction Rules

240 When building a PP/ST, or package using components from the FDP class, these construction rules will provide guidance on where to look and what to select from the class. Although they say “package”, this could be part of the construction of a PP/ST.

241

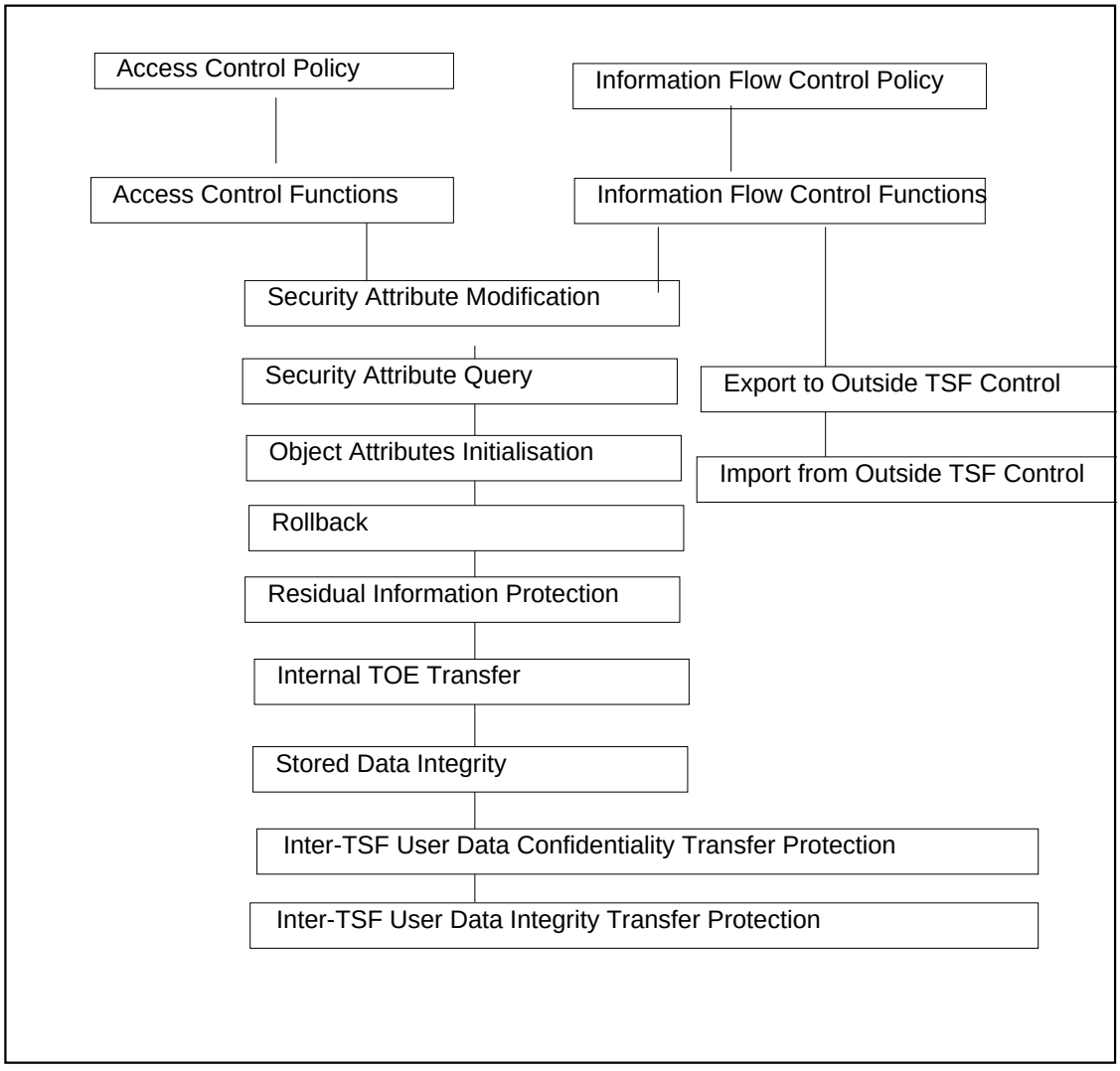

Figure 1.10 - User Data Protection Construction Rules

242 The requirements in the FDP class are defined in terms of a security function (abbreviated SF) which will implement a security function policy (SFP). Since a TOE may implement multiple SFPs simultaneously, the PP/ST author must specify the name for each SFP, so it can be referenced in other families. This name will then be used in each component selected to indicate that it is being used as part of the

definition of requirements for that function. This allows the author to easily indicate scope for operations such as objects covered, operations covered, authorized users, etc.

243 Each instantiation of a component applies to only one SFP. Therefore if a component in a PP or ST in one of its elements specify an SFP, this SFP will apply to all the elements in this component.

244 The key to selecting components from this family is to have a well defined TOE security policy to enable proper selection of the components from the two policy components; FDP_ACC and FDP_IFC. In FDP_ACC and FDP_IFC respectively, all access control policies and all flow control policies are named. Furthermore these components will define the subjects, objects and operations covered by this security function.

245 The following steps are guidance on how this class is applied in the construction of a PP/ST or a package describing the requirements for a security function:

a) Identify the policies to be enforced from the FDP_ACC, and FDP_IFC families. These families define scope of control for the policy, granularity of control and may identify some rules to go with the policy.

b) Identify the components and fill in any applicable operations in the policy components. The assignment operations may be filled in very generally (such as with a statement “All files”) or specifically (“The files “A”, “B”, etc.) depending upon the level of detail known.

c) Identify any applicable function components from the FDP_ACF and FDP_IFF families to address the respective policy families. Fill in the operations to make the components fit the requirements of the selected function envisioned or to be built.

d) Identify who will have the ability to control and change security attributes under the function, such as only a security administrator, only the owner of the object, etc. Select the appropriate components from FDP_SAM and fill in the operations. Refinements may be useful here to identify missing features such as that some or all changes must be done via trusted path.

e) Identify who will be able to query security attributes and select the appropriate components from FDP_SAQ.

f) Identify the source for initial values for new objects and subjects and select the appropriate components from the FDP_ACI family.

g) Identify any applicable rollback components from the FDP_ROL family.

h) Identify any applicable object reuse requirements from the FDP_RIP family.

i) Identify any applicable import or export components from the FDP_ITC and FDP_ETC families.

j) Identify any applicable internal TOE communication components from the FDP_ITT family.

k) Identify the requirements for integrity protection of stored information from the FDP_SDI.

l) Identify any applicable inter-TSF communication components from the FDP_UCT or FDP_UIT families.

FDP_ACC Access Control Policy

246 This family is based upon the concept of arbitrary controls on the interaction of subjects and objects. The scope and purpose of the controls is based upon the attributes of the accessor (subject), the attributes of the container being accessed (object), the actions (operations) and any associated access control rules. This family allows for both forms of access control normally considered "user-specified" and those considered "Security Administrator-specified"; for more information on this, see FDP_SAM Security Attribute Modification.

247 Examples of security policies that might satisfy these objectives are:

- Access Control Lists [Multics];
- Access Profiles [IBM RACF];
- Clark-Wilson TPs [Clark];
- Role-Based Access Controls [NIST];
- POSIX set-ID Mechanism [POSIX]; and
- Type Enforcement [LOCK].

User notes

248 This family defines an access control Security Function Policy (access control SFP) and the subjects, objects and operations that are covered by this Security Function Policy. The behaviour of this SFP will be defined by other families such as FDP_ACF, FDP_RIP.

249 The access control SFP covers a set of triplets subject, object, and operations. Therefore a subject can be covered by multiple SFPs but only with respect to a different operation or a different object. Of course the same applies to objects and operations.

250 This family would provide a PP/ST author the capability to specify several policies, for example, a fixed policy to be applied to one scope of control, and a flexible policy to be defined for a different scope of control.

251 A critical aspect of an access control SFP is that the PP/ST author must specify whether it can be modified. The ACC family does not address these aspects, such as who can change attributes, when they can change attributes and how they can change attributes. Some of these requirements are left undefined, but can be added as refinements, while others are covered elsewhere in other families and classes.

252 There are no audit requirements in FDP_ACC since this family specifies access control security function policy requirements. Audit requirements will be found in families specifying functions to satisfy the SFPs identified in this family.

253 This family can be applied multiple times in a PP/ST to different subsets of operations and objects. This will accommodate TOEs which contain multiple policies, each addressing a particular set of operations and objects. In other words, the PP/ST Author should specify the required information in the ACC component for each of the SFPs which the TOE will enforce. For example, a TOE incorporating three SFPs, each covering only a part of the objects, subjects, and operations within

the TOE, will contain one FDP_ACC.1 Subset Object Access Control component for each of the three SFPs making a total of three FDP_ACC.1 components.

Documentation notes

254 If AGD_USR User guidance or AGD_ADM Administrator guidance is applicable, either or both of these documents, as appropriate, should provide guidance with respect to each access control policy satisfying a FDP_ACC component. Documentation should be provided for end-users, authorised administrative users, or both, as appropriate for the nature of the objects and operations controlled by the policy.

FDP_ACC.1 Subset Object Access Control

User Application Notes

255 The terms object, and subject refer to generic elements in the TOE. For a policy to be implementable, the entities must be clearly identified. For a PP, the objects and operations might be expressed as classes: named objects, data repositories, observe accesses, etc. For a specific system these generic terms (subject, object) must be refined, e.g., files, registers, ports, daemons, open calls, etc.

256 This component simply specifies that the policy cover some well-defined set of operations on some subset of the objects. It places no constraints on any operations outside the set - including operations on objects for which other operations are controlled.

Documentation notes

257 The AGD_ADM Administrator guidance should define the subset of operations controlled by the SFP, describe the intended use of the operations, and provide a detailed rationale for the scope of the subset. The rationale should be sufficient to convince the evaluator that all listed objects are covered by the access control SFP. In the case of multiple SFPs, rationale should be provided to demonstrate that the SFPs do not conflict. If the PP/ST author claims that there is complete coverage of all objects and operations within the TSC, then rationale should be provided to demonstrate this as well as are no conflicts exist between the SFPs.

258 The ADV_FSP Functional specification should define the subset of operations and objects controlled by the policy, describe the intended use of these operations, and provide a detailed rationale for the scope of the subset. The ADV_FSP Functional specification should also be used to provide evidence for the rationale that the objects are covered by the access control SFPs and that there are no conflicts in the case of multiple SFPs.

Operations

Assignment:

259 **In FDP_ACC.1.1, assign the unique named access control SFP to be enforced by the TSF.**

260 **In FDP_ACC.1.1, list the set of combinations of subjects, objects, and operations covered by the SFP.**

FDP_ACC.2 Complete Object Access Control

User Application Notes

261 This component requires that all possible operations on objects, that are included in the SFP, are covered by an access control SFP.

262 The PP/ST author must demonstrate that each combination of objects and subjects is covered by an access control SFP.

Documentation notes

263 The AGD_ADM Administrator guidance should define the operations controlled by the policy, describe the intended use of the operations, and provide a detailed rationale for the scope of the subset. The rationale should be sufficient to convince the evaluator that all listed objects and operations are covered by the access control SFP.

264 The ADV_FSP Functional specification should, if in the PP/ST, be used to provide evidence for the rationale that all of the objects are covered by the access control SFPs.

Operations

Assignment:

265 In FDP_ACC.2.1, assign the unique named access control SFP to be enforced by the TSF.

266 **In FDP_ACC.2.1, list the set of combinations of subjects, and objects covered by the SFP.**

FDP_ACF Access Control Functions

267 This family describes specific functions that can implement the rules for access control SFPs. This family is dependent on the definition of a SFP.

User notes

268 This family provides a PP/ST author the capability to describe the rules for access control. Furthermore it provides requirements for the management of the access control attributes, such as changing the access control security attributes or explicitly disabling the modification of the access control attributes. The latter results in a fixed protection profile, such as can be found for systems where the access to objects will not change. An example of such an object is “Message of the Day”, which is readable by all, and changeable only by the authorised administrator.

269 If the TSF is in maintenance mode, the user is in a maintenance role, and if the TSF cannot enforce the security policy while in maintenance mode, then the TSP may be enforced by clearing the appropriate information, or by disabling the access mechanism, prior to entering maintenance mode.

270 There are no explicit components to specify other possible functions such as two-person control, sequence rules for operations, or exclusion controls. However, these mechanisms, as well as DAC and MAC mechanisms, can be represented with the existing components, by careful drafting of the access control rules.

271 A variety of acceptable access control functions may be specified in this family such as:

- Access control lists;
- Time-based access control specifications;
- Origin-based access control specifications;
- Owner-controlled access control attributes;
- Security Administrator-controlled access control attributes;
- Hierarchically controlled access control attributes;
- Hierarchical default access specifications;
- Per-subject default access specifications;
- Transferrable access control permissions; and
- Two-person control.

Documentation notes

272 User and administrative documentation should, as applicable, include information detailing what is the basis of mediation, what is the precedence of mediation when more than one conclusion could be reached given a set of attributes, etc.

273 The guidance documentation should describe the nature and scope of each access control policy and briefly describe the functions that implement the policy (FDP_ACF), the security attributes that govern the policy (FDP_SAQ, FDP_SAM), the initialisation rules for those attributes (FDP_ACI), and (if any) the default mechanisms for those attributes (FDP_ACI).

274 The guidance documentation should also provide guidance on the safe and effective use of the mechanisms.

FDP_ACF.1 Single Security Attribute Access Control

User Application Notes

275 This component provides requirements for a mechanism that mediates access control based on a single security attribute associated with subjects and objects. Each object and subject has a set of associated attributes, such as location, time of creation, access rights (such as ACL). This component allows the PP/ST author to specify the attribute that will be used for the access control mediation. Furthermore, this component allows access control rules, using this attribute, to be specified.

276 Examples of the attributes that a PP/ST author might assign are presented in the following paragraphs.

277 The *identity attribute* may be associated with users, subjects, or objects to be used for mediation. Examples of such attributes might be the name of the program image used in the creation of the subject, or a security attribute assigned to the program image.

278 The *time attribute* can be used to specify that access will be granted on Fridays only, during certain times of the day, or during a certain calendar year.

279 The *location attribute* could specify whether the location is the location of the request for the operation, the location where the operation will be carried out, or both. It could be based upon internal tables to translate the logical interfaces of the TSF into locations such as through terminal locations, CPU locations, etc.

280 The *grouping attribute* allows a single group of users to be associated with an operation for the purposes of access control. The maximum number of definable groups, the maximum membership of a group, and the maximum number of groups to which a user can concurrently be associated should be filled in by refinements, if required.

281 The *role attribute* may be used to define specific roles which users must assume to gain access to objects and operations. The maximum number of definable roles, the maximum size membership list of each role and the maximum number of roles in which a user can concurrently be acting should be filled in by refinement, if required.

282 A two-person rule can be implemented in FDP_ACF.1 by specifying a named group of “two users identities” in the assignment statement of FDP_ACF.1.2. This function will then mediate accesses to this named group based upon the rules defined in this component. A named group’s attributes are used just as a single attribute is used for mediation. Therefore, the two person rule will be enforced as both user identities will have to agree for an action to take place.

Operations

Assignment:

283 **In FDP_ACF.1.1, the PP/ST author should assign the *access control SFP name* which the TSF is to enforce.**

Assignment:

284 **In FDP_ACF.1.1, the PP/ST author should assign the *attribute* or *named group of attributes* that the function will use in the specification of the rules. The attributes may be user identity, subject identity, role, time of day, location, or any other attribute specified by the PP/ST author. The named group may be any group of attributes.**

Assignment:

285 **In FDP_ACF.1.2, the PP/ST author should assign *the SFP rules* which the TSF will enforce over the subjects and objects.**

FDP_ACF.2 Multiple Security Attribute Access Control

User Application Notes

286 This component provides requirements for a mechanism that mediates access control based on multiple security attribute associated with subjects and objects. Each object and subject has a set of associated attributes, such as location, time of creation, access rights (such as ACL). This component allows the PP/ST author to specify the attribute that will be used for the access control mediation. Furthermore, this component allows access control rules, using this attribute, to be specified.

287 A two-person rule can be implemented in FDP_ACF.2 by specifying a named group of "two users identities" in the assignment operation of FDP_ACF.2.1. This function will then mediate accesses to this named group based upon the rules defined in this component. A named group's attributes are used just as a single attribute is used for mediation. Therefore, the two person rule will be enforced as both user identities will have to agree for an action to take place.

Operations

Assignment:

288 In FDP_ACF.2.1, the PP/ST author should assign the *access control SFP name* which the TSF is to enforce.

Assignment:

289 In FDP_ACF.2.1, the PP/ST author should specify the ***multiple attributes or multiple named groups of attributes*** that the function will use in the specification of the rules. The attributes may be user identity, subject

identity, role, time of day, location, or any other attribute specified by the PP/ST author. The named group may be any group of attributes.

Assignment:

290 In FDP_ACF.2.2, the PP/ST author should assign *the SFP rules* which the TSF will enforce over the subjects and objects.

FDP_ACF.3 Access Authorisation

User Application Notes

291 This component provides requirements for the management of the access control security attributes. The access control security attributes must be able to explicitly grant access.

Operations

Assignment:

292 **In FDP_ACF.3.1, the PP/ST author should specify the *access control SFP name* being specified.**

FDP_ACF.4 Access Authorisation and Denial

User Application Notes

Operations

293 **In FDP_ACF.4.1, the PP/ST author should assign the *access control SFP name* being specified.**

FDP_ACF.5 Fixed Access Control

User Application Notes

294 This component ensures that the access control security attribute of a given SFP cannot be modified. Therefore subjects rights to the file cannot be changed. And in effect a static fixed access control policy is created

295 For example, the “message of the day” function typically provided by many multi user TOEs is covered by a fixed protection policy. The read and write operations can not be changed as they are built into the security function which provides only read access for users and only read/write access for administrators.

296 It is remarked that the user attributes and the object attributes could both control the access control between subjects and objects. It depends, for example, on whether an

Access Control List (Object attributes) or Capability Lists (Subject attributes) which set of attributes should be fixed.

Operations

Assignment:

297 **In FDP_ACF.5.1, the PP/ST author should specify the access control SFP name being specified.**

FDP_ACI Object Attributes Initialisation

298 This family defines the requirements on the initial and default values of access control security attributes. These rules address the need for objects to be protected appropriately by default.

User notes

299 The access control SFPs require as well object, user as subject attributes. The management requirements for the user and subject attributes are covered in FPT_FIA, therefore this family covers only the object attributes.

300 The default values of a parameter is the value the parameter would take when the parameter is instantiated without initial values. An initial value is provided during the instantiation (creation) of a parameter and is meant to override the default value.

301 This family allows the definition of default values, and their management. The last two components in this family provide restrictions on the values a parameter can take depending on the SFP, the attribute is being used for.

302 In the first three components, which address the default values and their management, the PP/ST author should define the inclination of the default values.: restrictive, meaning that the default value shall limit the capabilities of users as much as possible, permissive, meaning that the default value shall provide the users with the maximum set of capabilities within the SFP, or an other property., which must be refined by the PP/ST author. Only one of the properties may be chosen from the “selection” operation in FDP_ACI.x.1 since an attribute can not be permissive and restrictive at the same time.

303 As an example, if a set of requirements is projected on the FDP_ACI components, the following capabilities might be specified:
FDP_ACI.1 - umask capability must exist and can specify alternate default values;
FDP_ACI.2- admin can set umask and can specify alternate default values and modify values;
FDP_ACI.3 - user can set umask and can specify alternate default values and modify values;
FDP_ACI.4 - safe use setting value on create object (alternate default values);
FDP_ACI.5 - safe use on modifying value.

FDP_ACI.1 Static Attribute Initialisation

User Application Notes

304 This component requires that the TSF provide default values for relevant object security attributes, which can be overridden by an initial value, but that no mechanism need be supported for changing these defaults. It may still be possible for a new object to have different attributes at creation, if a mechanism exists to specify the permissions at time of creation.

Operations

Assignment:

305 **In FDP_ACI.1.1, list the *access control SFPs* for which the object security attributes are applicable.**

306 **In FDP_ACI.1.1, the PP/ST author should select whether the default property of the access control attribute will be *restrictive, permissive, or another property*. In case of another property the PP/ST author must refine this to a specific property.**

FDP_ACI.2 Administrator Defined Attribute Initialisation

User Application Notes

307 This component requires that there exists a default value for the object security attributes, which can be overridden by an initial value. Furthermore there must be an interface for a authorised administrative user to change the default values.

Operations

308 **In FDP_ACI.2.1, list the *access control SFPs* for which the object security attributes are applicable.**

309 **In FDP_ACI.2.1, the PP/ST author should select whether the default property of the access control attribute will be *restrictive, permissive, or another property*. In case of another property the PP/ST author must refine this to a specific property.**

FDP_ACI.3 User Defined Attribute Initialisation

User Application Notes

310 This component requires that the TOE support more flexible designation of users that can change initial values of object security attributes. Some may still be limited to security administrative users, some may be allowed to “owning” users, some may allow delegation chains, etc.

Operations

311 **In FDP_ACI.3.1, list the *access control SFPs* for which the object security attributes are applicable.**

312 **In FDP_ACI.3.1, the PP/ST author should select whether the default property of the access control attribute will be *restrictive, permissive, or another property*. In case of another property the PP/ST author must refine this to a specific property.**

FDP_ACI.4 Safe Access Control Attribute Initialisation

User Application Notes

313 This component ensures that the initial values must be valid with respect to the SFP, before they will be accepted as initial values for an object security attribute.

Operations

314 **In FDP_ACI.4.1, list the *access control SFPs* for which the object security attributes are applicable.**

FDP_ACI.5 Safe Access Control Attribute Modification

User Application Notes

315 This component ensures that modifications of the object security attributes value will only be accepted if the values are valid with respect to the SFP.

Operations

316 **In FDP_ACI.5.1, list the *access control SFPs* for which the object security attributes are applicable.**

FDP_ETC Export to Outside TSF Control

317 This family defines mechanisms for exporting information from the TOE such that the user data security attributes can be preserved. Consistency of these security attributes are addressed by FPT_SAC Security Attribute Consistency.

318 FDP_ETC is concerned with limitations on export, the form of the information (e.g., machine-readable, human-readable), user specification of security attributes, and association of security attributes with the exported information.

User notes

319 This family, and the corresponding Import family FDP_ITC, address how the TOE deals with information transferred into and outside its control, in principle it takes care of the import and export of the object security attributes. This can be to other TOEs without TSF such as printers or screens or to other TSFs.

320 A variety of activities might be involved here:

a) Reading a screen, exporting the displayed information;

b) Requesting a printout, exporting a printed representation;

c) Transferring information to a medium, without including any security attributes;

d) Transferring information, including security attributes, to a medium and handling it to enforce that the attributes are protected;

321 This family is not concerned with whether the information may be exported, it is concerned with under which conditions. If the information object is exported, this family ensures that the security attributes of the information is also exported and associated to the information in an accurate, unambiguous way.

322 The unambiguous association of the security attributes with the information can be achieved by physical means (the security attributes are on the same media), or by logical means (the security attributes are distributed differently but include a unique object identification, e.g., cryptographic checksum).

Update 1 Please review the paragraph below. Apparently I did something wrong. (I still it is a correct scheme though)

323 This family is concerned with exporting information and maintaining the association of security attributes as required by the SFP and resolving the address resolution with the exportation medium. Other families are concerned with other export aspects such as security attribute consistency, trusted channels, integrity, confidentiality, etc which are beyond the scope of this family. Furthermore, FDP_ETC is only concerned with the interface to the export medium. FDP_ITC is responsible for the other end point of the medium (the destination).

324 Some of the well know export requirements are:

a) export from a MLS TOE to a single level medium;

b) export from a single level TOE to a single level medium;

c) export from a single level or MLS TOE to a storage device; and

d) export from a single level or MLS TOE to a printer or CRT.

325 These export requirements may be handled by the TSF with or without human intervention as the depending on the IT limitations and the organisational security policy dictates. Regardless, these and other export requirements can be addressed by FDP_ETC.

326 Exporting from some environments (which may be a channel or medium) requires the TSF to translate the security attributes. This is subject of FPT_SAC In these cases, the TSF must have the list of security attributes for both environments and a translation table to be able to mediate access. If this is not available, human intervention may be required according to the SFP and the organisational SP.

327 To implement the export case (b) above, the TSF will have to resolve if the object level is permitted to be exported on the single level channel, satisfying FDP_ETC.x.1. In one case, it is a simple matter if the object level matches the channel level. In another case, the TOE and channel use different security attributes so that it requires the TSF to perform a translation to see if the object level and channel level are equivalent. These two cases both satisfy FDP_ETC.x.1.a to accurately and unambiguously represent the security attributes, in this case levels. Modifying the second case, It may be possible not to attach the security attributes to the object since the channel, being single level, already satisfies FDP_ETC.x.1.b to unambiguously associate the security attributes with the object being exported.

328 To simulate the TCSEC requirements, the elements will need refinements to restrict the scope of the specifications. FDP_ETC.x.1 uses similar wording as the TCSEC so that some additional rationale may be required in the PP/ST. FDP_ETC.x.1.b can be refined to require the security attributes to be of the same form and attached to the object being exported (such as in a file header) or for the security attributes to reside on the same media the object being exported (such as a floppy diskette). FDP_ETC.x.2 can be refined to restrict objects from being exported without the associated security attributes. Note that the security attributes encompass Bell and LaPadula labels as used in the TCSEC and some additional rationale or refinements may be required.

FDP_ETC.1 Export of User Data Without Security Attributes

User Application Notes

329 This component is used to specify the export of information without also transmitting security attributes.

Operations

330 **In FDP_ETC.1.1, the PP/ST author should assign the access control SFP or information flow control SFP name being specified.**

FDP_ETC.2 Export of User Data With Security Attributes

User Application Notes

331 The information is exported together with the security attributes. The security attributes are unambiguously associated to the information.

332 The information and the security attributes should be unambiguously be associated. This can be achieved by physically collocating the information and the security attributes, e.g., the same floppy, or using for example cryptographic techniques such as secure signatures to associate the attributes and the information.

Operations

333 **In FDP_ETC.2.4 the PP/ST author should assign any additional required SFP exportation control rules or "*none*" if there are no additional SFP capabilities.**

FDP_IFC Information Flow Control Policy

334 This family defines a set of information flow SFPs; and, for each, specifies the scope of control of the information flow SFP.

335 Examples of security policies that might satisfy this objective are:

- Bell and La Padula Security model [B&L]
- Biba Integrity model [Biba]
- Assured Pipelines [LOCK]
- Separation Kernel [Rushby]

User notes

336 The components in this class is capable of implementing the traditional Mandatory Access Control mechanisms. However, they are quire flexible, they allow the domain of flow control to be specified, and there is no requirement that the mechanism be based upon labels. The different strengths of the information flow control components also permit different degrees of exception to the policy.

337 Each SFP covers a set of triplets: subject, object, and operations. In the second component (FDP_IFC.2 Complete Information Flow Control), all operations between a subject and an object will be covered by the same SFP. Furthermore, each object will need to be covered by a SFP. Therefore for each action on an object there will be a set of rules that define whether this action is allowed. If there are multiple SFPs that are applicable for a given action, all involved SFPs must grant access for the action.

338 An information control policy covers a well-defined set of operations. The policy's coverage may be “complete” with respect to some object, or it may address only some of the operations that affect the object. A critical aspect of an information control policy is that it may be specified; that is, it is based upon some changeable attribute that determines the flow of the information.

339 Information Flow control Policies control access to the information which differs from access control policies which control access to the objects themselves. Since the policy security attributes are bound to the information and not the container, the attributes flow with the information from container to container.

340 Objects and operations can be expressed at multiple levels. If the PP/ST author has a specific system in mind (as might be the case for a Security Target), the operations might be at a system-specific level: files, open calls, etc. For a Protection Profile, the objects and operations might be expressed as classes: named objects, data repositories, observe accesses, etc.

341 The components in this class can be applied multiple times in a PP/ST to potentially to different subsets of operations and objects. This will accommodate TOEs which contain multiple policies, each addressing a particular set of objects, subjects, and operations.

Documentation notes

342 AGD_USR User guidance and AGD_ADM Administrator guidance should provide guidance with respect to each information flow control policy satisfying an FDP_IFC component.

343 The guidance documentation should describe the nature and scope of each information flow control policy. It should also provide guidance on the safe and effective use of the information flow security functions in the TSF.

344 The administrative guidance should describe the rationale for the scope of each information flow control policy.

345 There are no audit requirements in FDP_IFC since this family specifies information flow control security function policy requirements. Audit requirements will be found in families specifying functions to satisfy the SFPs identified in this family.

FDP_IFC.1 Subset Information Flow Control

User Application Notes

346 This component requires that an information flow control policy apply to a subset of the possible operations in the TOE.

Operations

347 **In FDP_IFC.1.1, the PP/ST author should assign the SFPs to be enforced by the TSF.**

348 **In FDP_IFC.1.1, the PP/ST author should assign the list of subjects and objects covered by the policy.**

349 **In FDP_IFC.1.1, the PP/ST author should assign the list of operations among subjects and objects covered by the SFP.**

FDP_IFC.2 Complete Information Flow Control

User Application Notes

350 This component requires that an information flow control policy apply to a subset of the possible operations in the TOE. Each object should be covered by at least one SFP.

Operations

Assignment:

351 **In FDP_IFC.2.1, the PP/ST author should assign the SFPs to be enforced by the TSF.**

352 **In FDP_IFC.2.1, the PP/ST author should assign the list of subjects and objects covered by the policy.**

FDP_IFF Information Flow Control Functions

353 This component specifies the requirements on function with respect to the information flow SFPs. It consists of two “trees:” one addressing the common information flow function issues, and a second addressing illicit information flow channels (i.e., covert channels) with respect to one or more information flow SFPs. This division arises because the issues concerning illicit information flows are, in some sense, orthogonal to the rest of an SFP. Illicit information flows are flows in violation of policy; thus they are not a policy issue (if they were explicitly allowed by the policy, they would not be illicit).

User notes

354 In order to implement strong protection against disclosure or modification in the face of untrusted software, controls on information flow are required. Access controls alone are not sufficient, because of the information flows implicit in controlled operations.

355 In this family, the phrase “types of illicit information flows” is used. This phrase may be used to refer to the categorization of flows as “Storage Channels” or “Timing Channels”, or it can refer to improved categorizations reflective of the state of the art.

356 The flexibility of the IFM components allow the definition of a privilege policy to allow controlled bypass of all or part of a particular SFP. If there is a need for a predefined approach to SFP bypass, the PP/ST author should consider incorporating a privilege policy as part of the IFM assignment.

Documentation notes

357 The documentation provided for AGD_ADM Administrator guidance should detail the bandwidth of the existing illicit information flows and any illicit information flows eliminated.

358 The documentation provided for AGD_USR User guidance and AGD_ADM Administrator guidance should detail the configuration, use and degree of protection provided by any user data exchange protection functions provided by the TSF.

FDP_IFF.1 Simple Security Attributes

User Application Notes

359 This component requires security attributes on containers of information, and on active recipients of information. It specifies the key rules that are enforced, and describes how security attributes are derived. It should be used when at least one of the information flow SFPs in the TSP is based on labels as defined in the Bell and LaPadula security policy model [Bell], but these security attributes do not form a hierarchy.

360 This component does not specify the details of how a security attribute is assigned (i.e., user versus process). Flexibility in policy is provided by having assignments that allow specification of additional policy and function requirements, as necessary.

361 Upon creation of a subject, the FPT_USB specifies that the object (image) and the user attributes determine the subject security attributes. If the SFP has additional rules on the management of the subject security attributes those can be specified under the additional information flow SFP rules. If there are specific rules for the object security attributes those can be specified under the additional information flow SFP rules.

Operations

362 **In FDP_IFF.1.1, the PP/ST author should list the information flow control SFPs enforced by the TSF.**

363 **In FDP_IFF.1.1 the PP/ST author should specify the minimum number and type of security attributes which the mechanism will enforce.**

364 **In FDP_IFF.1.2 the PP/ST author should for each operation, specify the security attribute-based relationship that must hold between subject and object security attributes if the operation is to be allowed.**

365 **In FDP_IFF.1.4 the PP/ST author should specify any additional information flow SFP rules or select none.**

FDP_IFF.2 Hierarchical Security Attributes

User Application Notes

366 This component requires that all information flow SFPs in the TSP use hierarchical security attributes that form a lattice.

367 It should be used when at least one of the information flow SFPs in the TSP is based on labels as defined in the Bell and LaPadula security policy model [Bell] and form a hierarchy.

Operations

368 **In FDP_IFF.2.1, the PP/ST author should list the information flow control SFPs enforced by the TSF.**

369 **In FDP_IFF.2.1 the PP/ST author should specify the minimum number and type of security attributes which the mechanism will enforce.**

370 **In FDP_IFF.2.2 the PP/ST author should for each operation, specify the security attribute-based relationship that must hold between subject and object security attributes if the operation is to be allowed.**

371 **In FDP_IFF.2.3 the PP/ST author should specify any additional information flow SFP rules or select none.**

FDP_IFF.3 Limited Illicit Information Flows

User Application Notes

372 This component should be used when at least one of the SFPs that requires control of illicit information flows does not require elimination of flows.

373 For the specified illicit information flows, certain maximum capacities are provided. In addition a PP/ST author has the ability to specify whether the illicit information flows must be audited.

Operations

Assignment:

374 **In FDP_IFF.3.1 the PP/ST author should list the information flow control SFPs enforced by the TSF.**

375 **In FDP_IFF.3.1 the PP/ST author should assign the maximum bandwidth permitted for any identified illicit information flows.**

376 **In FDP_IFF.3.1 the PP/ST author should list the types of illicit information flows which are subject to the maximum bandwidth limitation.**

FDP_IFF.4 Partial Elimination of Illicit Information Flows

User Application Notes

377 This component should be used when all the SFPs that requires control of illicit information flows require elimination of some (but not necessarily all) illicit information flows.

Operations

378 **In FDP_IFF.4.1 the PP/ST author should list the information flow control SFPs enforced by the TSF.**

379 **In FDP_IFF.4.1 the PP/ST author should assign the maximum bandwidth permitted for any identified illicit information flows.**

380 **In FDP_IFF.4.1 the PP/ST author should list the types of illicit information flows which are subject to the maximum bandwidth limitation.**

381 **In FDP_IFF.4.2 the PP/ST author should list the types of illicit information flows to be eliminated. This list may not be an empty list as this component requires that some illicit information flows are to be eliminated.**

FDP_IFF.5 No Illicit Information Flows

User Application Notes

382 This component should be used when all the SFPs that require control of illicit information flows require elimination of all illicit information flows.

Operations

383 **In FDP_IFF.5.1 the PP/ST author should list the information flow control SFPs enforced by the TSF.**

FDP_IFF.6 Illicit Information Flow Monitoring

User Application Notes

384 This component should be used when it is desired that the TSF provide the ability to audit the use of illicit information flows that exceed a specified capacity.

Operations

Assignment:

385 **In FDP_IFF.6.1 the PP/ST author should list the information flow control SFPs enforced by the TSF.**

Assignment:

386 **In FDP_IFF.6.1 the PP/ST author should provide the *list of types of illicit information flows* to be subjected to monitoring.**

Assignment:

387 **In FDP_IFF.6.1 the PP/ST author should provide the *specified capacity* that the TSF should monitor/audit illicit information flows against.**

Assignment:

388 **In FDP_IFF.6.1 the PP/ST author should provide the *list of types of illicit information flows* to be subjected to a maximum capacity.**

Assignment:

389 **In FDP_IFF.6.1 the PP/ST author should specify the *maximum capacity* below which all illicit information flows must be kept.**

FDP_ITC Import from Outside TSF Control

390 This family defines mechanisms for importing information from the TSC into to the TOE such that the user data security attributes can be preserved. Consistency of these security attributes are addressed by FDP_SAC.

391 FDP_ETC is concerned with limitations on import, the form of the information (e.g., machine-readable, human-readable), user specification of security attributes, and association of security attributes with the exported information.

User notes

392 This family, and the corresponding export family FDP_ETC, address how the TOE deals with information outside its control. This family is responsible for the assigning and abstraction of the object security attributes.

393 A variety of activities might be involved here:

a) Typing at a keyboard, importing the keyed-in information;

b) Importing information from an unformatted medium, without including any security attributes, and physically marking the medium to indicate it contents;

c) Importing information, including security attributes, from a medium and verifying that the object security attributes are appropriate;

d) Importing information, including security attributes, from a medium using a cryptographic sealing technique to protect the association of information and security attributes.

394 This family is not concerned whether the information may be imported. It is concerned with which values for the object security attributes to allocate.

395 There are two possibilities for the import of information: either the information is unambiguously associated with reliable object security attributes (values and meaning of the security attributes is not modified), or either no reliable security attributes or no security attributes at all are available. This class families with both cases.

396 If there are reliable security attributes available, they may have been associated with the information by physical means (the security attributes are on the same media), or by logical means (the security attributes are distributed differently, but include unique information identification, e.g., cryptographic checksum).

397 This family is concerned with importing information and maintaining the association of security attributes as required by the SFP and resolving the address resolution with the importation medium. Other families are concerned with other import aspects such as consistency, trusted channels, integrity, etc which are beyond the scope of this family. Furthermore, FDP_ITC is only concerned with the interface to the import medium. FDP_ETC is responsible for the other end point of the medium (the source).

398 Some of the well know import requirements are:

a) importing from a single level medium to a MLS TOE;

b) importing from a single level medium to a single level TOE;

c) importing from a storage device to a single level or MLS TOE; and

d) importing from a printer or CRT to a single level or MLS TOE.

399 These import requirements may be handled by the TSF with or without human intervention as the depending on the IT limitations and the organisational security policy. So, for example, if information is received on a “confidential” channel, the security attributes of the information will be set to “confidential”.

400 To implement the import case (b) above, the TSF will have to resolve if the information imported from the single level channel is permitted, satisfying FDP_ITC.1.2.a. In one case, it is a simple matter if the information level matches the channel level. In another case, the TOE and channel use different security attributes so that it requires the TSF to perform a translation to see if the object level and channel level are equivalent. These two cases both satisfy FDP_ITC.1.2.a to accurately and unambiguously represent the security attributes, in this case levels. Modifying the second case, It may be possible that there are no security attributes attached to the information since the channel, being single level, already satisfies FDP_ITC.1.2.a to unambiguously associate the security attributes with the information being exported.

401 To simulate the TCSEC requirements, the elements will need refinements to restrict the scope of the specifications. FDP_ITC.1.2 uses similar wording as the TCSEC so that some additional rationale may be required in the PP/ST. FDP_ITC.1.2 can be refined to require the security attributes to be of the same form and attached to the object being imported (such as in a file header) or for the security attributes to reside on the same media the object being imported (such as a floppy diskette). FDP_ITC.1.2 can be refined to restrict information from being imported without the associated security attributes. Note that the security attributes encompass Bell and LaPadula labels as used in the TCSEC and some additional rationale or refinements may be required.

FDP_ITC.1 Import of Reliable Objects Controlled Under an Access Control Policy

User Application Notes

402 This component is to be specified once for each access control SFP named in FDP_ACC.x.1 that has SFP-specific requirements regarding import of objects from outside of the TSC.

Operations

Refinement:

403 **FDP_ITC.1.1.a may be further refined to require the security attributes to be attached to the information being imported.**

404 **FDP_ITC.1.1.a may be further refined to require the security attributes to reside on the same medium as the information being imported.**

FDP_ITC.2 FDP_ITC.1 Import of User Data Without Security Attributes

User Application Notes

405 This component is to be specified once for each access control SFP named in FDP_ACC.x.1 that has SFP-specific requirements regarding importation of objects from outside of the TSC.

406 This component provides for import of information from outside the TSC to inside the TSC in the case of absence of reliable security attributes. These attributes can be provided by an authorised user (mandatory) or by automated rules (to be specified by PP/ST author).

Operations

Refinement:

407 **FDP_ITC.2.1.a may be further refined to require the security attributes to be attached to the information being imported.**

408 **FDP_ITC.2.1.a may be further refined to require the security attributes to reside on the same media the information being imported.**

FDP_ITT Internal TOE Transfer

409 This family provides requirements that address protection of user data when it is transferred between parts of a TOE across a channel. This may be contrasted with the FDP_UCT/FDP_UIT family, which provides protection for user data when it is transferred between distinct TSFs across an external channel.

User notes

410 The requirements in this family allow a PP/ST author to specify the desired security for user data while in transit within the TOE. This security could be protection against disclosure, loss of integrity, or loss of availability.

411 The determination of the degree of physical separation above which this family should apply depends on the intended environment of use. In a hostile environment, there may be risks arising from transfers between parts of the TOE separated by only a system bus. In more benign environments, the transfers may be across more traditional network media.

412 Refinement of “an approved method” allows a PP/ST author to specify a particular approach to the confidentiality or integrity protection, for example, physically secured lines or cryptographic solutions.

Evaluator notes

413 Based on technology available at the time of the development of this document, the only practical mechanism available to a TSF to provide this protection are cryptography-based.

FDP_ITT.1 Basic Internal Transfer Protection

Operations

Assignment:

414 **In FDP_ITT.1.1, the PP/ST author should specify the *access control SFP, information flow control SFP* covering the information being transferred.**

Selection:

415 **In FDP_ITT.1.1 the PP/ST author should select the protection the user data should have while in transport. The options are *disclosure, modification, non-availability*.**

FDP_ITT.2 Transmission Separation by Attribute

User Application Notes

416 One of the ways to achieve separation of channels based on SFP-relevant attributes is through the use of distinct encryption algorithms.

417 For example, this component could be used to provide different protection to information with different clearance levels.

Operations

Assignment:

418 In FDP_ITT.2.1, the PP/ST author should specify the *access control SFP, information flow control SFP* covering the information being transferred.

Selection:

419 In FDP_ITT.2.1 the PP/ST author should select the protection the user data should have while in transport. The options are *disclosure, modification, non-availability*.

Assignment:

420 **In FDP_ITT.2.2, the PP/ST author should assign the *security attributes that require separate transmission channels*.**

FDP_ITT.3 Integrity Monitoring

User Application Notes

421 This component is used in combination with either FDP_ITT.1 or FDP_ITT.2. It ensures that the TSF checks received user data (and their attributes) for integrity.

422 The PP/ST author has to specify which types of errors must be detected. The PP/ST author should consider: modification of data, substitution of data, unrecoverable ordering change of data, replay of data, incomplete data, in addition to other integrity errors.

423 The PP/ST author must specify which actions the TSF should take on detection of a failure. For example: ignore the user data, request the data again, inform the authorised administrator, reroute traffic for other lines.

Operations

Selection:

424 **In FDP_ITT.3.1, the PP/ST author should select the *access control SFP, information flow control SFP*.**

Assignment:

425 **In FDP_ITT.3.1, the PP/ST author should identify the type of possible *integrity errors* to be monitored for during transmission of the user data.**

Assignment:

426 **In FDP_ITT.3.2, the PP/ST author should identify the *action to be taken* by the TSF when an integrity error is encountered.**

FDP_ITT.4 Attribute-Based Integrity Monitoring

427 This component is used in combination with FDP_ITT.2. It ensures that the TSF checks received user data (and their attributes) for integrity.

428 For example, this component could be used to provide different protection to information with different integrity levels such as high integrity required.

429 The PP/ST author has to specify which types of errors must be detected. The PP/ST author should consider: modification of data, substitution of data, unrecoverable ordering change of data, replay of data, incomplete data, in addition to other integrity errors.

430 The PP/ST author should specify which attributes require a different transmission channel.

431 The PP/ST author must specify which actions the TSF should take on detection of a failure. For example: ignore the user data, request the data again, inform the authorised administrator, reroute traffic for other lines.

Operations

Selection:

432 In FDP_ITT.4.1, the PP/ST author should select the *access control SFP, information flow control SFP*.

Assignment:

433 In FDP_ITT.4.1, the PP/ST author should identify the type of possible *integrity errors* to be monitored for during transmission of the user data.

Assignment:

434 **In FDP_ITT.4.1, the PP/ST author should provide a list of *security attributes that require separate transmission channels.***

Assignment:

435 In FDP_ITT.4.2, the PP/ST author should identify the *action to be taken* by the TSF when an integrity error is encountered.

FDP_RIP Residual Information Protection

436 This family addresses the need to ensure that deleted information is no longer accessible, and that newly-created objects do not contain information from previously used objects within the TOE. This family does not address objects stored off-line.

User notes

437 This family requires protection for information that has been logically deleted (not available to the user but still within the system and may be recoverable) or released, but may still be physically present within the TOE. In particular, this includes information that is contained in an object, as part of the TSF reusable resources, where destruction of the object does not necessarily equate to destruction of the resource or any contents of the resource.

438 The TSF controls access to information that is usually not part of any currently defined or accessible object; however, in certain cases this may not be true. For example, if object "A" is deleted, which resides on a disk defined to be an object, the information from object "A" is under the control of FDP_RIP even though it is still part of a defined object (i.e., the disk object).

439 It is important to note that FDP_RIP applies only to on-line objects and not off-line objects such as those backed-up on tapes. For example, if a file is deleted in the TOE, FDP_RIP.4 will enforce that no residual information exists upon deallocation; however, the TSF cannot extend this enforcement to that same file which exists on the off-line back-up. Therefore that same file is still available.

440 FDP_RIP and FDP_ROL conflict in that FDP_RIP.2 and FDP_RIP.4 ensure that the contents will be made unavailable at the time the application releases the object to the TSF. Therefore, these two components can not be used with FDP_ROL since there would be no information to roll back. FDP_RIP.1 and FDP_RIP.3 may be used with FDP_ROL since these two components specify that the information must be made unavailable at the time it is reallocated so that the information is present in the TOE for FDP_ROL to make use of.

441 There are no audit requirements in FDP_RIP because this is not a user-invokable function. Auditing of allocated or deallocated resources would be auditable as part of the access control SFP or the information flow control SFP operations.

442 This family should cover the objects specified in the access control SFP or the information flow control SFP as specified by the PP/ST author.

FDP_RIP.1 Subset Residual Information Protection on Allocation

User Application Notes

443 This component requires that, for a subset of the objects in the TOE, the TSF will ensure that there will be no available residual information contained in a resource allocated to those objects.

Operations

444 **In FDP_RIP.1.1, the PP/ST author should assign the *list of objects* subject to residual information protection on allocation.**

FDP_RIP.2 Subset Residual Information Protection on Deallocation

User Application Notes

445 This component requires that, for a subset of the objects in the TOE, the TSF will ensure that there will be no residual information contained in a resource when it is deallocated.

Operations

Operation : No permitted operation.

446 **In FDP_RIP.2.1, the PP/ST author should assign the *list of objects* subject to residual information protection on deallocation.**

FDP_RIP.3 Full Residual Information Protection on Allocation

User Application Notes

447 This component requires that, for **all objects** in the TOE, the TSF will ensure that there will be no available residual information contained in a resource allocated to those objects.

FDP_RIP.4 Full Residual Information Protection on Deallocation

User Application Notes

448 This component requires that, for **all objects** in the TOE, the TSF will ensure that there will be no residual information contained in a resource when it is deallocated.

FDP_ROL Rollback

449 This family addresses the need to return to a well defined valid state , for example the need of a user to undo keystrokes in an editor or to undo transactions in case of an incomplete series of transaction as in the case of databases.

User notes

450 This family is intended to assist a user in returning to a well defined valid state after the user decided that he wanted the last set of actions undone, or, for example in distributed databases, the return of both databases to the situation before an operation failed.

451 FDP_RIP and FDP_ROL conflict in that FDP_RIP.3 and FDP_RIP.4 enforces that the contents will be made unavailable at the time the application releases the object to the TSF. Therefore, these two components can not be used with FDP_ROL since there would be no information to roll back. Only FDP_RIP.1 and FDP_RIP.2 may be used with FDP_ROL since these two components specify that the information must be made unavailable at the time it is reallocated so that the information is present in the TOE for FDP_ROL to make use of it.

452 The rollback requirement is bounded by certain limits. For example a texteditor only allows roll-back upto a certain number of commands. Another example would be reverting to backups. If backup tapes are roulated, after a tape is reused, the information can no longer be retrieved. This also poses a bound on the roll-back.

Documentation notes

453 AGD_USR User guidance and AGD_ADM Administrator guidance should provide guidance use rollback and how to manage rollback.

FDP_ROL.1 Basic Rollback

User Application Notes

454 This component allows a user or subject to undo a set of operations on a predefined set of objects.

455 The undo is only possible within certain limits, for example upto a number of characters or upto a time limit.

Operations

456 **In FDP_ROL.1.1 the PP/ST author should assign the SFP.**

457 **In FDP_ROL.1.2 the PP/ST author should specify the list of users and/ or subjects that are authorised to invoke the roll-back capability.**

458 **In FDP_ROL.1.2 the PP/ST author should specify the list of operations that can be reverted.**

459 **In FDP_ROL.1.2 the PP/ST author should specify the objects which are subjected to the roll-back policy.**

460 **In FDP_ROL.1.3 the PP/ST author should assign the boundary in which rollback operations may be performed. The boundary may be specified as a predefined period of time, for example, operations may be undone within the past two minutes. Other possible boundaries may be defined as the maximum number of operations allowable or the size of a buffer.**

FDP_ROL.2 Advanced Rollback

User Application Notes

461 This component enforces that the TSF provide the capability to rollback all operations; however, the user can choose to rollback only a part of them.

Operations

462 **In FDP_ROL.2.1 the PP/ST author should assign the SFP.**

463 **In FDP_ROL.2.2 the PP/ST author should specify the list of users and/ or subjects that are authorised to invoke the roll-back capability.**

464 **In FDP_ROL.2.2 the PP/ST author should specify the objects which are subjected to the roll-back policy.**

465 **In FDP_ROL.2.3 the PP/ST author should assign the boundary in which rollback operations may be performed. The boundary may be specified as a predefined period of time, for example, operations may be undone within the past two minutes. Other possible boundaries may be defined as the maximum number of operations allowable or the size of a buffer.**

FDP_ROL.3 Administrative Rollback

User Application Notes

466 This component provides the authorised administrator the capability to change the boundaries on the rollback, within the limits of the TOEs capabilities.

Operations

Assignment:

467 **In FDP_ROL.2.1 the PP/ST author should assign the SFP.**

468 **In FDP_ROL.3.2 the PP/ST author can specify the the boundary in which the authorised administrator may change the boundaries. The boundary may be specified as a predefined period of time, for example, operations may be undone within the past two minutes. Other possible boundaries may be defined as the maximum number of operations allowable or the size of a buffer.**

FDP_SAM Security Attribute Modification

469 This family defines the rules for setting and modifying values of security attributes according to the user data protection policy.

User notes

470 The rules of applying these functions may be different for different attributes (e.g., an access control mechanism might be based both on object ownership and an access control list, but the rules for changing ownership and the access control list may be different). This should be specified by using refinements as required.

471 In an environment where the TOE is made up of multiple physically-separated parts that form a distributed system, the timing issues with respect to propagation of attribute modification become very complex, especially if the attributes are replicated between the parts of the TOE. In such situations, use of components from FPT_TRC, TSF Data Replication Consistency, is advisable.

FDP_SAM.1 Minimal Attribute Modification

User Application Notes

472 At this level, only a Security Administrator can modify the security attributes associated with the function.

Operations

473 **In FDP_SAM.1.1 the PP/ST author should assign the SFP name.**

474 **In FDP_SAM.1.1 the PP/ST author should assign the list of security attributes that the security administrators can modify.**

FDP_SAM.2 Basic Attribute Modification

User Application Notes

475 At this level, authorised users and Security Administrator can modify the security attributes for objects, users and subjects.

Operations

476 **In FDP_SAM.1.1 the PP/ST author should assign the SFP name.**

477 **In FDP_SAM.1.1 the PP/ST author should assign the list of security attributes that the security administrators and authorised users can modify.**

FDP_SAM.3 Basic Attribute Modification (Ref: Safe Attribute Modification)

User Application Notes

478 This component ensures that the changes being affected in SAM.1 and SAM.2 are valid with respect to the SFP.

Operations

Assignment:

479 **In FDP_SAM.1.1 the PP/ST author should assign the SFP name.**

480 **In FDP_SAM.1.1 the PP/ST author should assign the list of security attributes that will be checked on their validity before being changed.**

FDP_SAQ Security Attribute Query

481 This family addresses the need for authorized users and subjects operating on behalf of authorized users, to query security attributes. Each security function should include aspects that satisfy these components.

FDP_SAQ.1 Minimal Attribute Query

User Application Notes

482 This component provides capabilities to the authorised administrator to change security attributes.

Operations

483 **In FDP_SAQ.1.1, the PP/ST author should assign the SFP name.**

484 **In FDP_SAQ.1.1 the PP/ST author should assign the list of security attributes that can be modified by the authorised administrator.**

FDP_SAQ.2 User Attribute Query

User Application Notes

485 This component provides capabilities to the authorised administrator to change security attributes. Furthermore a user can view certain security attributes.

Operations

486 **In FDP_SAQ.2.1, the PP/ST author should assign the SFP name.**

487 **In FDP_SAQ.2.1 the PP/ST author should assign the list of security attributes that can be modified by the authorised administrator.**

FDP_SDI Stored Data Integrity

488 This family provides requirements that address protection of user data while it is stored within the TOE.

User notes

489 Hardware glitches errors may affect data stored in memory. This family provides requirements to detect these unintentional errors. User data integrity while stored on storage devices within the TSC are also addressed by this family.

490 If a subject actively might try to modify the data, not this family but IFF or ACF is required.

491 This family differes from FDP_ITT Internal TOE Transfer which protects the user data from integrity errors while being transferred within the TOE.

FDP_SDI.1 Stored Data Integrity Monitoring

User Application Notes

492 This component protects data stored on media against integrity errors. The PP/ST author can specify which action should be taken in case of an integrity fault.

Operations

Assignment:

493 **In FDP_SDI.1.1 the PP/ST author should specify the objects covered by the SDI policy.**

494 **In FDP_SDI.1.2 the PP/ST author, if applicable, should specify the other integrity errors against which stored data must be protected. Examples of possible integrity errors are: modification of data, incomplete data.**

495 **In FDP_SDI.1.3 the PP/ST author should specify the actions to be taken in case of an integrity failure.**

FDP_SDI.2 Stored Data Attribute-Based Integrity Monitoring

User Application Notes

496 This component protects data stored on media against integrity errors. The PP/ST author can specify which action should be taken in case of an integrity fault.

Operations

Assignment:

497 **In FDP_SDI.2.1 the PP/ST author should specify the list of objects covered by the SDI policy.**

498 **In FDP_SDI.2.2 the PP/ST author, if applicable, should specify the other integrity errors against which stored data must be protected. Examples of possible integrity errors are: modification of data, incomplete data.**

499 **In FDP_SDI.1.3 the PP/ST author should specify the actions to be taken in case of an integrity failure.**

FDP_UCT Inter-TSF User Data Confidentiality Transfer Protection

500 This family defines the mechanism requirements for ensuring the confidentiality of user data when it is transferred using a channel between distinct TOEs or users on distinct TOEs. Confidentiality is enforced by preventing unauthorised disclosure of data in transit between the two end points. The end points may be a TSF or a user.

User notes

501 This family provides a requirement for the protection of user data during transit. In contrast FTP_ITC handles TSF data.

Documentation notes

502 The documentation provided for AGD_USR User guidance and AGD_ADM Administrator guidance should detail the configuration, use and degree of protection provided by any user data exchange protection mechanisms provided by the TSF.

FDP_UCT.1 Basic Data Exchange Confidentiality

User Application Notes

503 The TSF has the ability to protect from disclosure some user data which is exchanged. The protection may only be to the level of a group of TOEs and is not necessarily to the individual level.

Operations

504 **In FDP_UCT.1.1, the PP/ST author should specify the SFP for which requirements are being created.**

505 **In FDP_UCT.1.1, the PP/ST author should specify whether this element applies to a mechanism transmitting or receiving objects.**

FDP_UIT Inter-TSF User Data Integrity Transfer Protection

506 This family defines the mechanism requirements for ensuring the integrity of user data when it is transferred using an external channel between distinct TOEs or users on distinct TOEs. . Integrity is enforced by preventing unauthorised modification of data in transit between the two end points. The end points may be a TSF or a user.

User notes

507 This family deals with the protection of user data from unauthorised modification when it is transferred using an external channel between distinct TOEs or users on distinct TOEs; while FPT_ITC handles TSF data.

508 FDP_UIT and FDP_UCT are duals of each other as FDP_UCT addresses user data. confidentiality. Therefore, this same mechanism could possibly be used to implement other families such as FDP_UCT and FDP_ITC.

Documentation notes

509 The documentation provided for AGD_USR User guidance and AGD_ADM Administrator guidance should detail the configuration, use and degree of protection provided by any user data exchange protection mechanisms provided by the TSF.

FDP_UIT.1 Basic Data Exchange Integrity

User Application Notes

510 The TSF has a basic ability to send data in a manner protected from modification such that the receiving party is able to identify that a modification has occurred. There is no requirement for a TSF mechanism to attempt to recover from the modification.

Operations

511 **In FDP_UIT.1.1, the PP/ST author should specify the SFP for which requirements are being created.**

512 **In FDP_UIT.1.2, the PP/ST author should specify whether this element applies to a TSF transmitting or receiving objects.**

Selection:

513 **In FDP_UIT.1.2 the PP/ST author should select whether the data should be protected from modification, deletion, insertion or replay.**

514 **In FDP_UIT.1.3 the PP/ST author should select whether the errors of the type: modification, deletion, insertion or replay are detected.**

FDP_UIT.2 Destination Data Exchange Recovery

User Application Notes

515 This component provides the ability to recover from a set of identified transmission errors.

Operations

516 **In FDP_UIT.2.1, assign the TSF for which requirements are being created.**

517 **In FDP_UIT.2.1, select the type of integrity errors that the TSF will be able to recover the original user data.**

FDP_UIT.3 Source Data Exchange Recovery

User Application Notes

518 This component provides the ability to recover from a set of identified transmission errors, if required with the help of the other TSF.

Operations

519 **In FDP_UIT.3.1, assign the SFP for which requirements are being created.**

520 **In FDP_UIT.3.1, select the type of integrity errors that the TSF, with the help of the source TSF, to be able to recover the original user data.**

References

521 [NIST] Ferraiolo, D., Cugini, J., and Kuhn, R.D., *Role Based Access Control: Features and Motivations*, Proceedings of the Ninth Annual Computer Security Applications Conference, December 1995.

522 [Clark] Clark, D. D., and D. R. Wilson, *A Comparison of Commercial and Military Security Policies*, Proceedings of the IEEE Symposium on Security and Privacy: pp. 184 - 194, Oakland, CA., April 1987.

523 [Biba] Biba, K. J., *Integrity Considerations for Secure Computer Systems*, ESD-TR-372, ESD/AFSC, Hanscom AFB, Bedford Mass., April 1977.

524 [Bell] Bell, D. E. and LaPadula, L. J., *Secure Computer System*s, ESD-TR-73-278, Volume I-III, MITRE Corp., Bedford Mass., November 1972 - June 1974.

525 [IBM RACF] National Computer Security Center, Final Evaluation Report of IBM MVS/XA with RACF version 1.8", CSC-EPL-88/003, 15 June 1988.

526 [POSIX] Inf. Tech. Port. O.S. Sys. Interface, Part 1: System Application Programming Interface, IEEE STD 1003.1-1990, ISO STD ISO/IEC 9545-1

- Access Control Lists [Multics]
- Access Profiles [IBM RACF]
- Clark-Wilson TPs [Clark]
- Role-Based Access Controls [NIST]
- POSIX set-ID Mechanism [POSIX]
- Type Enforcement [LOCK]
- Bell and La Padula Security model [ref. to B&L paper]
- Biba Integrity model [ref. to Biba paper]
- Assured Pipelines [ref. to LOCK paper]
- Separation Kernel [ref. to Rushby paper]

Class FIA

Identification and Authentication

527 A common security requirement is to control the access of users to the TOE. This involves not only establishing the claimed identity of each user, but also verifying that each user is indeed who he/she claims to be. This is achieved by requiring users to provide the TSF with some information that is known by the TSF to be associated with the user in question.

528 Families in this class address the requirements for functions to establish and verify a claimed user identity. Identification and Authentication is required to ensure that users are associated with the proper Security Attributes (e.g., identity, groups, roles, security or integrity levels).

529 The unambiguous identification of authorised users and the correct association of security attributes with users and subjects is critical to the enforcement of the security policies.

530 The FIA_UID family addresses determining the identity of a user.

531 The FIA_UAU family addresses verifying the identity of a user.

532 The FIA_ADA and FIA_ADP families address the administration and protection of authentication data.

533 The FIA_AFL family addresses defining limits on repeated unsuccessful authentication attempts.

534 The FIA_ATD and FIA_ATA families address the definition and administration of user attributes that are used in the enforcement of the TSP.

535 The FIA_USB family addresses the correct association of security attributes for each authorised user.

536 The FIA_SOS family addresses the generation and verification of secrets that satisfy a defined metric.

537 Other classes of requirements (e.g., User Data Protection, Security Audit) are dependent upon correct identification and authentication of users in order to be effective.

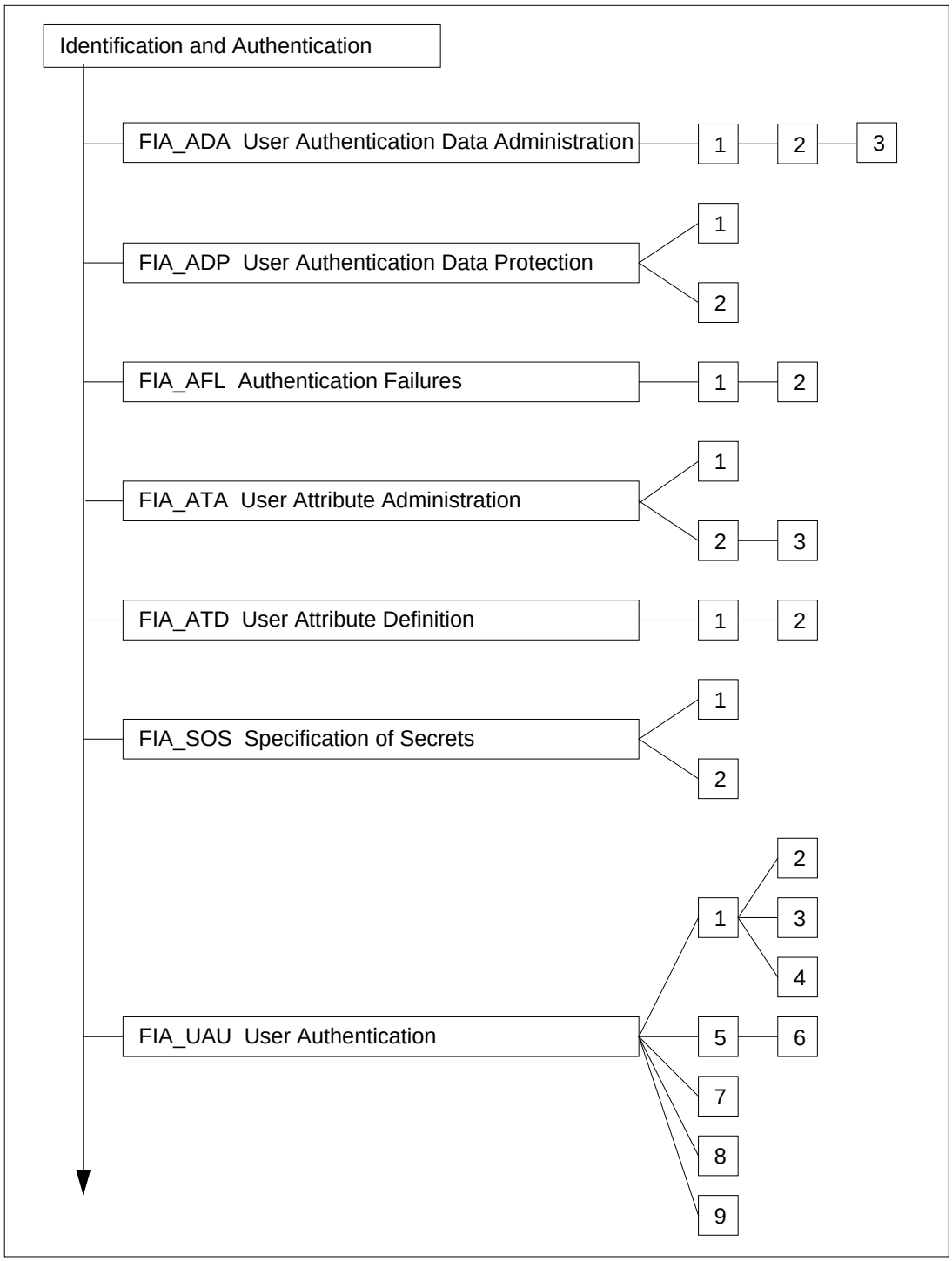

Figure 1.11 - Identification and Authentication class decomposition

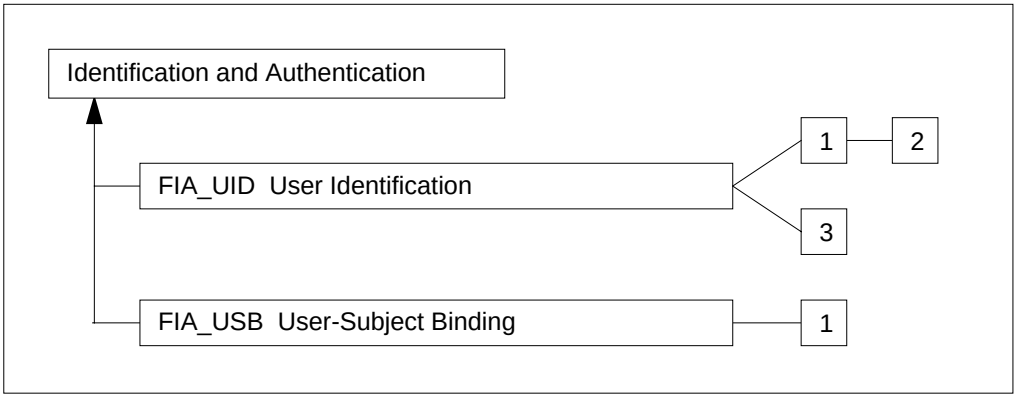

Figure 1.12 - Identification and Authentication class decomposition (Cont.)

Construction Rules

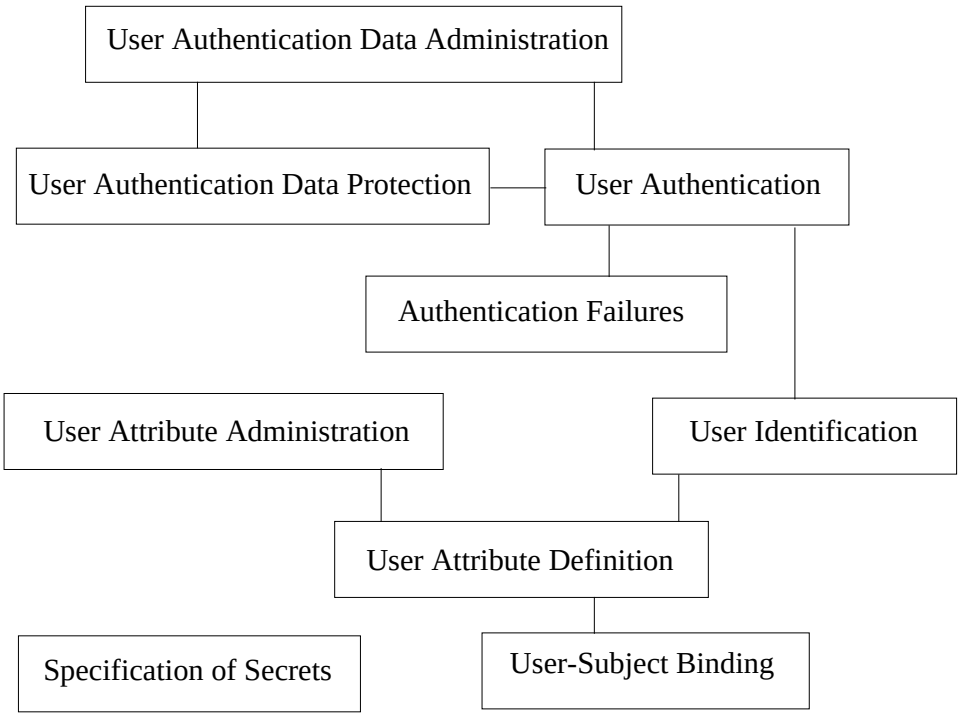

Figure 1.13 - Identification and Authentication requirements construction rules

538 When building a PP, ST, or package using components from the FIA class, these construction rules will provide guidance on where to look and what to select from the class.

539 Selecting a component in the FIA_ADA family the PP/ST author shall define:

- the authentication mechanisms to be used,

540 and requests:

- functions for authorised administrators ro perform functions to initialise and modify user authentication data related to the authentication mechanisms used,
- functions for an authorised user to modify their own authentication data related to the authentication mechanisms used.

541 Selecting a component in the FIA_ADP family the PP/ST author requests:

- protection of the raw form of authentication data and stored authentication data.

542 Selecting a component in the FIA_AFL family the PP/ST author defines:

- the number of unsuccessful authentication attempts that are allowed before the user session establishment procedure is terminated, and
- whether it is the user account or point of entry that is disabled when that number is reached and the condition for re-enabling whatever was disabled.

543 Selecting a component in the FIA_ATA family the PP/ST author requests:

- the initialisation of user attributes with provided default values;
- the ability for authorised administrators to modify any user's security attributes; and
- the ability of an authorised user to modify their own user attributes,

544 and defines:

- whether the ability to display or modify user attributes should be provided by the TSF and, if so, to whom.

545 Selecting a component in the FIA_ATD family the PP/ST author requests:

- that user security attributes necessary to enforce the TSP be associated with groups of users or uniquely associated with each individual user.

546 Selecting a component in the FIA_SOS family the PP/ST author defines:

- a metric to be used by a TSF-provided mechanism to verify the quality of a secret; and
- a quality metric to be used by a TSF-provided mechanism when generating secrets.

547 Selecting a component in the FIA_UAU family the PP/ST author requests:

- an authentication mechanism to authenticate a user's claimed identity that operates with re-usable authentication data, and/or
- an authentication mechanism to authenticate a user's claimed identity that operates with single-use authentication data, and/or
- an authentication mechanism to authenticate a user's claimed identity that detects and prevents the use of forged or copied authentication data, or
- multiple authentication mechanisms to authenticate a user's claimed identity;
- the enforcement of the use of separate authentication mechanisms to authenticate specific authentication events;
- the ability to restrict the actions a user can perform before their identity is authenticated; and
- the ability to install an authentication mechanism into the TSF.

548 and defines:

- if relevant, the number of different authentication mechanisms to be provided and a list of what they are;
- if relevant, which authentication mechanism is to be used to authenticate which specific authentication event;
- if relevant, conditions that require re-authentication;
- if relevant, actions that a user can perform before they are authenticated; and
- if relevant, whether an installed authentication mechanism is to be used in place of or in addition to any of the existing authentication mechanisms.

549 Selecting a component in the FIA_UID family the PP/ST author requests:

- the ability to identify a user;
- the ability to restrict the actions that a user can perform before they are identified to the TSF; and

550 and defines:

- actions that users can perform before they are identified.

551 Selecting a component in the FIA_USB family the PP/ST author requests:

- association of user security attributes with subjects, that the user owns, in accordance with the TSP.

FIA_ADA User Authentication Data Administration

552 This family defines requirements to initially set up or change user authentication data.

User notes

553 Authentication data is the data users must provide when authenticating themselves. Authentication data can be something the user knows (e.g., password, secret), possesses (e.g., smart card, one time password generator), or is (e.g., fingerprint, palmprint, retinal scan pattern).

Documentation notes

554 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the TSF authentication data administration mechanism. [ADV_FSP Functional specification]

b) Guidance on using the TSF authentication data administration mechanism. [AGD_ADM Administrator guidance]

FIA_ADA.1 User Authentication Data Initialisation

User Application Notes

555 In this component, only the authorised administrator is permitted to initialise user authentication data used to authenticate users.

Operations

Assignment:

556 **In FIA_ADA.1.1, the PP/ST author must define the *authentication mechanisms* used to authenticate the user for which authentication data has to be initialised.**

FIA_ADA.2 Basic User Authentication Data Administration

User Application Notes

557 In this component, only the authorised administrator is permitted to initialise and modify user authentication data used to authenticate users.

Operations

Assignment:

558 In FIA_ADA.2.1, the PP/ST author must define the *authentication mechanisms* used to authenticate the user for which authentication data has to be initialised **and can be modified**.

FIA_ADA.3 Expanded User Authentication Data Administration

User Application Notes

559 In this component, only the authorised administrator is permitted to initialise and modify any user's authentication data related to the authentication mechanisms used to authenticate the user. In addition, users are authorised to modify their own authentication data.

Operations

Assignment:

560 In FIA_ADA.3.1, the PP/ST author must define the *authentication mechanisms* used to authenticate the user for which authentication data has to be initialised and can be modified.

FIA_ADP User Authentication Data Protection

561 To establish the claimed identity of a user, the TOE will use information provided by the user, and known by the TOE to be associated with the user in question. This family defines the requirements to protect this information against unauthorised access or modification. It includes requirements to assure the integrity of, or prevent the unauthorised use of, authentication data.

User notes

562 A PP/ST author may choose to explicitly specify via a refinement operation which users are authorised to observe, modify and/or destroy the authentication data protected by the TSF. For example: administrators are authorised to observe, modify, and destroy all authentication data, while users are only able to modify (change but not observe) their own authentication data. As another example: administrators are authorised to observe, modify and destroy all authentication data, while users may observe all authentication data in an encrypted form but may only modify their own authentication data.

FIA_ADP.1 Basic User Authentication Data Protection

User Application Notes

563 Permanent storage of authentication data refers to the form of the authentication data that the TSF maintains and against which it compares the authentication data provided by the user requesting authentication (e.g., stored passwords, digital representation of biometric data, challenge/response information). This stored version of the authentication data may be protected in many ways, for example, via hashing or encryption functions, or stored in the clear and protected via access controls or privilege mechanisms.

Documentation notes

564 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the manner by which the user authentication data is protected from unauthorised use, observation, modification and destruction while the authentication data is stored in the TSF. [ADV_FSP Functional specification]

FIA_ADP.2 Extended User Authentication Data Protection

User Application Notes

565 The raw form of authentication data is the form that the data is in when it is provided across the TSFI for authentication, such as an unencrypted password. Another example of raw authentication data could be the sequence of bits that represent a

biometric signature if there is a possibility to insert that sequence of bits across the TSFI. The raw form must have a potential for reuse.

566 The sole use of this raw form of authentication data is for authenticating the user. Because there is no other authorised use of this data, the requirement for protection of this data calls for protection against access even by administrative users who may be able to override access controls. Examples of the types of access that must be prevented include perusing memory for buffers that may contain raw user authentication data. The raw form of the authentication data shall be available only to the TSF authentication mechanism(s).

Documentation notes

567 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the manner by which the user authentication data is protected from unauthorised use, observation, modification and destruction **at all times while it is under TSF control**. [ADV_FSP Functional specification]

FIA_AFL Authentication Failures

568 This family addresses requirements for defining default values for authentication attempts and TSF actions in cases of authentication attempt failure. Parameters include, but are not limited to, the number of attempts and time thresholds.

Documentation notes

569 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the number of unsuccessful authentication attempts allowed before termination of the session establishment procedure and the manner in which it is specified by the authorised administrator. [ADV_FSP Functional specification]

b) Guidance on how to use the TSF provided functions related to administration of authentication attempt failures. [AGD_ADM Administrator guidance]

FIA_AFL.1 Basic Authentication Failure Handling

User Application Notes

570 It is acceptable for the number of unsuccessful authentication attempts to be specified by the TOE developer. It is also acceptable if this value is also modifiable by a user authorised to perform administrative functions. The unsuccessful authentication attempts need not be consecutive, but rather the count should be from the last successful session establishment.

571 It is acceptable for the conditions upon which the user session establishment procedure can be re-enabled to be specified by the TOE developer. It is also acceptable if these conditions are also modifiable by a user authorised to perform administrative functions.

572 TOEs usually ensure that there is at least one user account that cannot be disabled in order to prevent denial of service. In order to accomplish this for such accounts as these and points of entries like the console, the condition for re-enabling the session establishment procedure could be a zero or very small time-out value that must expire.

Operations

Assignment:

573 **In FIA_AFL.1.1, the PP/ST author must provide a *number* to specify how many unsuccessful authentication attempts that are acceptable before the TSF terminates the procedure to establish a user session.**

Selection:

574 **In FIA_AFL.1.2, the PP/ST author must specify whether it is the *user account and/or the point of entry* (e.g., workstation, port) that is to be disabled when the defined number of unsuccessful authentication attempts is reached.**

Assignment:

575 **In FIA_AFL.1.2, the PP/ST author must provide the *conditions upon which the user session establishment procedure can be re-enabled*. These conditions may include a timeout value that must elapse or the explicit reset of the disabled account or point of entry by the authorised administrator.**

FIA_AFL.2 Basic Authentication Failure Handling

User Application Notes

576 It is acceptable for the number of unsuccessful authentication attempts to be specified by the TOE developer. It is also acceptable if this value is also modifiable by a user authorised to perform administrative functions. The unsuccessful authentication attempts need not be consecutive, but rather the count should be from the last successful session establishment.

577 It is acceptable for the conditions upon which the user session establishment procedure can be re-enabled to be specified by the TOE developer. It is also acceptable if these conditions are also modifiable by a user authorised to perform administrative functions.

578 TOEs usually ensure that there is at least one user account that cannot be disabled in order to prevent denial of service. In order to accomplish this for such accounts as these and points of entries like the console, the condition for re-enabling the session establishment procedure could be a zero or very small time-out value that must expire.

Operations

Assignment:

579 In FIA_AFL.1.1, the PP/ST author must provide a *number* to specify how many unsuccessful authentication attempts that are acceptable before the TSF terminates the procedure to establish a user session.

Selection:

580 In FIA_AFL.1.2, the PP/ST author must specify **the choices that the TSF must provide to the authorised administrator** on the disabling of *user accounts and/or points of entry* (e.g., workstation, port) that is to be disabled when the defined number of unsuccessful authentication attempts is reached.

Assignment:

581 In FIA_AFL.1.2, the PP/ST author must provide the *conditions upon which the user session establishment procedure can be re-enabled*. These conditions may include a timeout value that must elapse or the explicit reset of the disabled account or point of entry by the authorised administrator.

FIA_ATA User Attribute Administration

582 All authorised users have a set of attributes to support the enforcement of the TSP including attributes to support the authentication security policy. This family defines the requirements to initially set up, change, or review this collective set of user attributes. This family defines requirements to enable new user identities to be added, and old user identities to be removed, modified or invalidated.

Documentation notes

583 The indicated assurance documentation (if applicable) shall contain the following information:

a) Guidance on how to use the TSF provided user attribute administration functions related to user-manageable attributes. [AGD_USR User guidance].

b) Guidance on how to use the TSF provided user attribute administration functions. [AGD_ADM Administrator guidance].

c) Specification of the TSF provided user attribute administration functions. [ADV_HLD High-level design, ADV_FSP Functional specification].

d) Specification of the default values for user attributes [ADV_FSP Functional specification].

FIA_ATA.1 User Attribute Initialisation

User Application Notes

584 At this level, the TSF must provide a function to initialise user attributes and default values for these attributes. It is acceptable for the default values to be specified by the TOE developer. It is also acceptable if these default values are modifiable by a user authorised to perform administrative functions.

FIA_ATA.2 Basic User Attribute Administration

User Application Notes

585 At this level, the TSF must provide a mechanism to allow the display and modification of user attributes. Only an authorised administrator is permitted to modify user security attributes.

Operations

Selection:

586 **In FIA_ATA.2.1, the PP/ST author must specify whether the TSF shall provide the ability to *display and/or modify* user attributes.**

FIA_ATA.3 Extended User Attribute Administration

User Application Notes

587 This component can be used to permit users to modify some of their own attributes.

Operations

Selection:

588 In FIA_ATA.3.1, the PP/ST author must specify whether the TSF must provide the ability to *display and/or modify* user attributes.

FIA_ATD User Attribute Definition

589 All authorised users may have a set of security attributes, other than the user's identity, that is used to enforce the TSP. This family defines the requirements for associating user security attributes with users as needed to support the TSP.

User notes

590 There are dependencies on the individual security policy definitions. These individual definitions should contain the listing of attributes that are necessary for policy enforcement.

Documentation notes

591 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the user-related TSP attributes and the manner in which they are associated with the user. [ADV_FSP Functional specification]

FIA_ATD.1 User Attribute Definition

User Application Notes

592 At this component level, it is acceptable for more than one user to share the same association with TSP attributes. There can be a many-to-one mapping of user identities to user attribute definitions.

FIA_ATD.2 Unique User Attribute Definition

User Application Notes

593 At this component level it is **not** acceptable for more than one user to share the same association with TSP attributes. While is it possible for different users to have identical user attribute values in their user attribute definition, there must be a one-to-one mapping between user identities and user attribute definitions.

FIA_SOS Specification of Secrets

594 This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets and generate secrets to satisfy the defined metric. Examples of such mechanisms may include: automated checking of user supplied passwords, automated password generation, etc.

User notes

595 Secrets are the authentication data provided by the user for an authentication mechanism that is based on knowledge the user possesses.

FIA_SOS.1 Selection of Secrets

User Application Notes

596 Secrets can be generated by the user. This component ensures that those user generated secrets can be verified to meet a certain quality metric.

Operations

Assignment:

597 **In FIA_SOS.1.1, the PP/ST author must provide *a defined quality metric*. The quality metric specification can be as simple as a description of the quality checks to be performed or as formal as a reference to a government published standard that defines the quality metrics that secrets must meet. Examples of quality metrics could include a description of the alphanumeric structure of acceptable secrets and/or the space size that acceptable secrets must meet.**

FIA_SOS.2 TSF Generation of Secrets

User Application Notes

598 When a pseudo-random number generator is used in a secret generation algorithm, it should accept as input random data that would provide output which has a high degree of unpredictability. This random data (seed) can be derived from a number of available parameters such as a system clock, system registers, date, time, etc. The parameters should be selected to ensure that the number of unique seeds that can be generated from these inputs should be at least equal to the minimum number of secrets that must be generated.

Operations

Assignment:

599 **In FIA_SOS.2.1, the PP/ST author must provide *a defined quality metric*. The quality metric specification can be as simple as a**

description of the quality checks to be performed or as formal as a reference to a government published standard that defines the quality metrics that secrets must meet. Examples of quality metrics could include a description of the alphanumeric structure of acceptable secrets and/or the space size that acceptable secrets must meet.

600 **In FIA_SOS.2.2, the PP/ST author must provide a *list of TSF functions* for which the TSF generated secrets must be used. An example of such a function could include a password based authentication mechanism.**

FIA_UAU User Authentication

601 This family defines the types of user authentication mechanisms supported by the TSF. This family defines the required attributes on which the user authentication mechanisms must be based.

Documentation notes

602 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the TSF authentication mechanism(s). [ADV_FSP Functional specification]

b) Guidance on use of the TSF authentication mechanism(s). [AGD_USR User guidance]

c) Guidance on configuring the TSF authentication mechanism(s). [AGD_ADM Administrator guidance]

FIA_UAU.1 Basic User Authentication

User Application Notes

603 This component addresses requirements for authentication mechanisms based on reusable authentication data. Reusable authentication data can be something the user is (e.g., fingerprints, palmprints), something the user possesses (e.g., smart cards, one-time pads), or, more commonly, something the user knows (e.g., passwords).

FIA_UAU.2 Single-use Authentication Mechanisms

User Application Notes

604 This component addresses requirements for authentication mechanisms based on single-use authentication data. Single-use authentication data can be something the user has or knows, but not something the user is. Examples of single-use authentication data includes such things as single-use passwords, encrypted time-stamps, random numbers from a secret lookup table.

FIA_UAU.3 Integrity of Authentication

User Application Notes

605 This component addresses requirements for authentication mechanisms which provide integrity of authentication data. An authentication mechanism that is subject to integrity is one where it can be demonstrated that the authentication data used by a user could not have been forged by the user or copied from any other user

including trusted users of the TOE. This mechanism provides confidence that users authenticated by the TSF are actually who they claim to be. This component may only be useful with authentication mechanisms which are based on authentication data that cannot be shared (e.g., biometrics). It is impossible for a TSF to detect or prevent the sharing of passwords outside the control of the TSF.

FIA_UAU.4 Multiple Authentication Mechanisms

User Application Notes

606 The use of this component allows specification of requirements for more than one authentication mechanism. For each separate mechanism, requirements must be chosen from components FIA_UAU.1, FIA_UAU.2, FIA_UAU.3, and FIA_UAU.9 to be applied to each mechanism. It is possible that the same component could be selected twice, but in doing so it would have to be selected twice with respect to two different security functions within the TOE.

Operations

Assignment:

607 **In FIA_UAU.4.1, the PP/ST author must define the *number* of required authentication mechanisms. The number specified must be greater than one for the component to be differentiated from FIA_UAU.1.**

Assignment:

608 **In FIA_UAU.4.1, the PP/ST author must provide a *list of different mechanisms* for authentication. For each of the listed authentication mechanisms, the mechanism must meet a requirement chosen from:**

- **FIA_UAU.1 Basic User Authentication**
- **FIA_UAU.2 Single-use Authentication Mechanisms**
- **FIA_UAU.3 Integrity of Authentication**
- **FIA_UAU.9 Installable Authentication Mechanisms**

FIA_UAU.5 Policy-based Authentication Mechanisms

User Application Notes

609 This component requires that a policy be defined that describes when each of the required authentication mechanisms shall be employed with respect to each authentication event, such as modifying specific user security attributes. The policy is specified by the TOE developer and cannot be modified by a user authorised to perform administrative functions.

610 This component also allows different authentication mechanisms to be specified for different methods and modes of access to the TOE (e.g., reusable passwords for directly connected terminals, one-time passwords for dial-in access). For each

separate mechanism, requirements must be chosen from components FIA_UAU.1, FIA_UAU.2, FIA_UAU.3, and FIA_UAU.9 to be applied to each mechanism. It is possible that the same component could be selected twice, but in doing so it would have to be selected twice with respect to two different security functions within the TOE.

Operations

Assignment:

611 **In FIA_UAU.5.1, the PP/ST author must define the *number* of required authentication mechanisms.**

Assignment:

612 **In FIA_UAU.5.1, the PP/ST author must provide a *list of different mechanisms* for authentication. For each of the listed authentication mechanisms, the mechanism must meet requirements chosen from:**

- **FIA_UAU.1 Basic User Authentication**
- **FIA_UAU.2 Single-use Authentication Mechanisms**
- **FIA_UAU.3 Integrity of Authentication**
- **FIA_UAU.9 Installable Authentication Mechanisms**

Refinement:

613 **In FIA_UAU.5.2, the PP/ST author must define the *separate authentication mechanisms for specific authentication events*.**

FIA_UAU.6 Configurable Authentication Mechanisms

User Application Notes

614 This component requires that an authorised administrator be allowed to define a policy that describes when each of the required authentication mechanisms shall be employed with respect to each authentication event, such as authenticating specific security attributes.

615 This component also allows different authentication mechanisms to be specified for different methods and modes of access to the TOE (e.g., passwords for directly connected terminals, one-time passwords for dial-in access). For each separate mechanism, requirements must be chosen from components FIA_UAU.1, FIA_UAU.2, FIA_UAU.3, and FIA_UAU.9 to be applied to each mechanism. It is possible that the same component could be selected twice, but in doing so it would have to be selected twice with respect to two different security functions within the TOE.

Operations

Assignment:

616 In FIA_UAU.6.1, the PP/ST author must define the *number* of required authentications mechanisms.

Assignment:

617 In FIA_UAU.6.1, the PP/ST author must provide a *list of different mechanisms* for authentication. For each of the listed authentication mechanisms, the mechanism must meet requirements chosen from:

- FIA_UAU.1 Basic User Authentication
- FIA_UAU.2 Single-use Authentication Mechanisms
- FIA_UAU.3 Integrity of Authentication
- FIA_UAU.9 Installable Authentication Mechanisms

Refinement:

618 In FIA_UAU.6.2, the PP/ST author must define the *separate authentication mechanisms for specific authentication events*.

Refinement:

619 **In FIA_UAU.6.3, the PP/ST author must provide the mapping associating the *separate authentication mechanisms with specific authentication events*. Examples of such mapping may include the requirement for a password based authentication mechanism for local user login but a one-time authentication mechanism for login via a remote machine.**

FIA_UAU.7 On-demand Authentication

User Application Notes

620 This component addresses potential needs to re-authenticate users at FSP defined points in time. These may include user requests for the TSF to perform security relevant actions, as well as requests from non-TSF entities for re-authentication (e.g., a server application requesting that the TSF re-authenticate the client it is serving).

Operations

Assignment:

621 **In FIA_UAU.7.1, the PP/ST author shall specify the *list of conditions requiring re-authentication*. This list could include a specified user inactivity period that has elapsed, the user has requested a change in active security attributes, or the user has requested the TSF to perform a security critical function.**

FIA_UAU.8 Timing of Authentication

User Application Notes

622 This component requires that the PP/ST author define the TSF-mediated actions that can be performed by the TSF on behalf of the user before the claimed identity of the user is authenticated. The TSF-mediated actions should have no security considerations with users incorrectly identifying themselves prior to being authenticated. For all other TSF-mediated actions not in the list, the user must be authenticated before the action can be performed by the TSF on behalf of the user.

Operations

Assignment:

623 **In FIA_UAU.8.1, the PP/ST author must specify a *list of TSF-mediated actions* that can be performed by the TSF on behalf of a user before the claimed identity of the user is authenticated. An example of such an action might include the request for help on the login procedure.**

FIA_UAU.9 Installable Authentication Mechanisms

User Application Notes

624 This component is intended to allow the installation of new authentication mechanisms, such as token-based cards, biometrics, or other trusted third-party mechanisms that may be used in place of or addition to any existing authentication mechanisms. This component can be combined with any other component from this family.

Operations

Selection:

625 **In FIA_UAU.9.2, the PP/ST author must specify whether the installable mechanism is to be *in place of or in addition to* any existing authentication mechanism.**

FIA_UID User Identification

626 This family defines the conditions under which users shall be required to identify themselves before performing any other actions that are to be mediated by the TSF and which require user identification.

Documentation notes

627 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the user identification function. [ADV_FSP Functional specification]

b) Guidance to users on how to identify themselves to the TOE. [AGD_USR User guidance]

c) Guidance to administrators on how to define users. [AGD_ADM Administrator guidance]

FIA_UID.1 Basic User Identification

User Application Notes

628 In this component, it is acceptable for more than one user to share the same identity.

629 This component is always needed. It provides the basic component of user identification needed in order for a security policy to be implemented.

FIA_UID.2 Unique Identification of Users

User Application Notes

630 In this component, users are not allowed to share the same identity. The TSF must ensure that each user is assigned a unique identity for the purpose of user accountability.

FIA_UID.3 Timing of Identification

User Application Notes

631 This component requires that the PP/ST author define the TSF-mediated actions that can be performed by the TSF on behalf of the user before requiring that the user present their identity. For all other TSF-mediated actions not in the list, the user must provide their identity before the action can be performed by the TSF on behalf of the user.

Operations

Assignment:

632 **In FIA_UID.3.1, the PP/ST author must define the *list of actions* that can be performed by the TSF on behalf of the user before the user is identified. The PP/ST author must determine, for each action considered to be listed in this operation, as to whether user identification should be required before that action is allowed. An example of an action that might be allowed is for the user to request help on the identification and authentication process.**

FIA_USB User-Subject Binding

633 An authenticated user, to perform an action in the TOE, typically activates a subject. The user's security attributes are associated (totally or partially) with this subject. This family defines requirements to create and maintain the association of the user's security attributes to a subject acting on the user's behalf.

User notes

634 There are dependencies on the individual security policy definitions (e.g., from class FDP). These individual definitions should contain the listing of the user security attributes that should be bound to subjects. The identification of those attributes can be found in the definition of the policy in the components related to that policy.

Documentation notes

635 The indicated assurance documentation (if applicable) shall contain the following information:

a) Specification of the manner in which user attributes are associated with subjects that the user owns. [ADV_FSP Functional specification]

FIA_USB.1 User-Subject Binding

User Application Notes

636 The phrase "acting on behalf of" has proven to be a contentious issue in the previous criteria. It is intended that a subject is acting on behalf of the user who caused the subject to come into being. Therefore, when a subject is created as a result of the identification and authentication process, that subject is acting on behalf of the user who was identified and authenticated. If that same subject behaves as a server to process arbitrary requests received from other users, the subject is still acting on behalf of the user who was identified and authenticated. If a user creates such a server subject, that user must assume accountability for the operations performed by the server, even if the operation is the result of another user's requesting the server to perform it.

Class FPR

Privacy

637 This class is based on the current available knowledge about Privacy techniques. Since research in this area is still on going, in the future these components might need expansion or revision.

638 This class describes the requirements that could be levied to satisfy the users' privacy needs, while still allowing the system flexibility as far as possible to maintain sufficient control over the operation of the system.

639 In the components of this class the authorised administrators (set of all authorised administrator roles) is mentioned as being covered by the required security functions. However, if there is an administrator with authorisations specific to the privacy functions, that administrator is excluded from the set of authorised administrators to whom the requirements apply, although not so indicated in the components.

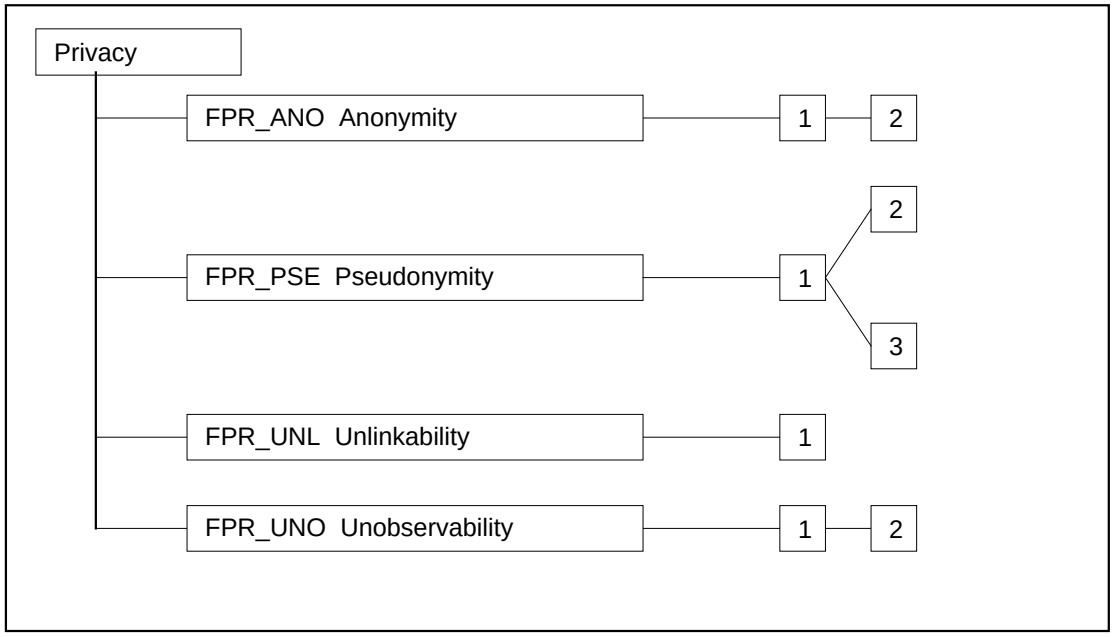

Figure 1.14 - Privacy class decomposition

640 This class, together with other classes, such as those concerned with audit, access control, trusted path, and non-repudiation provides the flexibility to specify the

desired privacy behaviour. On the other hand, the requirements in this class might pose limitations on the use of the components of other classes such as FIA, FAU. For example if authorised administrators are not allowed to see the user identity (e.g. Anonymity or Pseudonymity), the audit requirements are seriously impaired, since the auditor is not allowed to know who is responsible for a security relevant event.

641 This class describes four families: Anonymity, Pseudonymity, Unlinkability and Unobservability. Anonymity, Pseudonymity and Unlinkability have a complex interrelationship. From a privacy point of view, anonymity is the strongest requirement, no identity information is provided whatsoever. Followed in strength by unlinkability, which states that no relationship between operations can be found. However, the potential exists that a unique reference to the user is provided. The weakest variant is pseudonymity. A unique alias can be provided, but each transaction might employ the same alias. Since the relationships between operations could be derived, this might allow for profiling of the user.

642 All families assume that a user does not explicitly perform an action that discloses the user's own identity. Therefore, the TSF is, for example, not expected to screen the user name in electronic messages or databases.

643 All families in this class have components that can be scoped through the operations. The operations allow to state the number of cooperating users/subjects to which the TSF must be resistant, and whether authorised administrators (e.g. the audit authorised administrator, or the I&A authorised administrator) are included or excluded from this set. An example of an instantiation of anonymity could be: "The TSF shall ensure that two cooperating users and/or subjects, excluding authorised administrators, are unable to determine the user identity bound to the teleconsulting application".

FPR_ANO Anonymity

644 Anonymity ensures that a subject may use a resource or service without disclosing its user identity.

User notes

645 The intention of this family is to specify that a user or subject might take action without releasing its user identity to others such as users, subjects, or objects.

646 Therefore if a subject, using anonymity, performs an action, another subject will not be able to determine either the identity or even a reference to the identity of the user employing the subject. The focus of the anonymity is on the protection of the users identity, not on the protection of the subject identity. Therefore the identity of the subject is not protected from disclosure.

647 Although the identity of the subject is not released to other subjects or users the TSF is not explicitly prohibited from obtaining the users identity. In case the TSF is not allowed to know the identity of the user, FPR_ANO.2 could be invoked. In that case the TSF should not request the user information.

648 The interpretation of “determine” should be taken in the broadest sense of the word. The PP/ST author might want to use a Strength of Function to indicate how much rigour should be applied.

649 The component levelling distinguishes between the users and an authorised administrator. An authorised administrator is often excluded from the component and therefore allowed to retrieve a users identity. However there is no specific requirement that an authorised administrator must be able to have the capability to determine the users identity.

650 Although some systems will provide anonymity for all services which are provided, other systems only provide anonymity for certain subjects/operations. To provide this flexibility an operation is included where the scope of the requirement is presented. If the PP/ST author wants to address all subjects/operations, the words “All subjects and all operations” could be provided.

651 Possible applications include the ability to make enquiries of a confidential nature to public databases, respond to electronic polls, or make anonymous payments or donations.

652 Examples of potential hostile users or subjects are providers, system operators, communication partners and users, who smuggle malicious parts, (e.g. Trojan Horses) into systems. All of these users can investigate usage patterns, (e.g., which users used which services) and misuse this information.

FPR_ANO.1 Anonymity

User Application Notes

653 This component ensures that the identity of a user is protected from disclosure.

Operations

Assignment:

654 **In FPR_ANO.1.1 the PP/ST author should specify the [*set of users and/ or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies 'a single user or subject', the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.**

Selection:

655 **In FPR_ANO.1.1 the PP/ST author should specify whether authorised administrators (responsible for other functions) are included or excluded from the scope. If the PP/ST author specifies 'included' the set of subjects/users may include authorised administrators, but not the authorised administrator responsible for privacy.**

Assignment:

656 **In FPR_ANO.1.1 the PP/ST author should identify the [*list of subjects and/or operations*] where the user identity of the subject should be protected, for example "the voting application".**

FPR_ANO.2 TSF Anonymity

User Application Notes

657 This component is used to prohibit the TSF from accepting any user-identity related information.

Operations

Assignment:

658 In FPR_ANO.2.1 the PP/ST author should specify the [*set of users and/or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies 'a single user or subject', the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.

Selection:

659 In FPR_ANO.2.1 the PP/ST author should specify whether authorised administrators (responsible for other functions) are included or excluded from the scope. If the PP/ST author specifies 'included' the set of subjects/ users may include authorised administrators, but not the authorised administrator responsible for privacy.

Assignment:

660 In FPR_ANO.2.1 the PP/ST author should identify the [*list of subjects and/ or operations*] where the users identity of the subject should be protected, for example “the voting application”.

Assignment:

661 **For FPR_ANO.2.2 the PP/ST author should identify the [*list of subjects*] where the users identity of the subject should be protected, for example the voting application.**

Assignment:

662 **For FPR_ANO.2.2 the PP/ST author should identify the [*list of operations*] where the users identity should be protected, for example “the accessing of job descriptions”.**

FPR_PSE Pseudonymity

663 Pseudonymity ensures that an entity may use a resource or service without disclosing its identity, but can still be accountable for that use. The user can be accountable through directly being related to a reference (alias) held by the TSF, or by providing an alias which will be used for processing purposes such as an account number.

User notes

664 In several respects pseudonymity resembles anonymity. Both pseudonymity and anonymity protect the identity of the user, but in pseudonymity a reference to the users identity is maintained for accountability or other purposes. Of course, although not explicitly stated in the component, this alias cannot be related to the users identity without cooperation of the TSF.

665 The component FPR_PSE.1 does not specify the requirements on the alias. For the purpose of specifying requirements on this reference two sets of requirements are presented: FPR_PSE.2 and FPR_PSE.3.

666 A way to use the reference is by being able to obtain the original user identifier. For example in a digital cash environment it would be an advantage to be able to trace the users identity when a check has been issued multiple times (i.e. fraud). In general the users identity needs to be retrieved under specific conditions. The PP/ ST author might want to incorporate FPR_PSE.2 Reversible Pseudonymity to describe those services instead.

667 Another usage of the reference is as an alias for a user. For example a user does not wish to be identified, but can provide an account to which the resource utilisation should be charged. In those cases the reference to the user identity is an alias for the user where other users or subjects can use the alias for performing their functions without ever obtaining the users identity (for example statistical operations on use of the system). In this case the PP/ST author might wish to incorporate FPR_PSE.3 Alias Pseudonymity to specify the rules to which the reference must conform.

668 Using these constructs above, digital money can be created using FPR_PSE.2 Reversible Pseudonymity. In FPR_PSE.2 Reversible Pseudonymity will specify that the user identity will be protected and, if so specified in the condition, there can be a requirement to trace the user identity if the digital money is spent twice. Thereby when the user is honest, the user identity is protected, and if the user tries to cheat, the user identity can be traced.

669 A different kind of system could be a digital credit card, where the user will provide a pseudonym which indicates an account from which the cash can be substracted. In that case for example FPR_PSE.3 Alias Pseudonymity could be used. FPR_PSE.3 Alias Pseudonymity will specify that the user identity will be protected and, furthermore this component will specify that the same user will only get assigned values for which he/she has provided money (if so specified in the conditions).

670 It should be realised that especially the more stringent components potentially cannot be combined with other requirements, such as identification and authentication or audit.

671 The interpretation of “determine the identity” should be taken in the broadest sense of the word. The information is not provided by the TSF during the operation, nor can the entity determine the subject or the owner of the subject that invoked the operation, nor will the TSF record information, available to the users or subjects, which might release the user identity in the future.

672 It is emphasised that the reference to the users identity shall not be traceable to the users identity without cooperation of the TSF. It should also be interpreted as that the identity of the subject cannot be revealed if this would compromise the identity of the user. Which information is considered to be sensitive depends on the effort an attacker is capable of spending. Therefore the FPR_PSE Pseudonymity family is subject to Strength of Function requirements.

673 Possible applications include the ability to charge a caller for premium rate telephone services without disclosing his or her identity, or to be charged for the anonymous use of an electronic payment system.

674 Examples of potential hostile users are providers, system operators, communication partners and users, who smuggle malicious parts, e.g. Trojan Horses into systems. All of these attackers can investigate which users used which services and misuse this information. Additionally to Anonymity services Pseudonymity Services contain methods for authorisation without identification, especially for anonymous payment (“Digital Cash”). This helps providers to get their payment in a secure way while maintaining customer anonymity.

FPR_PSE.1 Pseudonymity

User Application Notes

675 This component provides the user protection against disclosure of its identity to other users. The user will remain accountable for its actions.

676 This component is dependent on either FPR_PSE.2 or FPR_PSE.3. However, these other components could be located in a separate TOE.

Operations

Assignment:

677 **In FPR_PSE.1.1 the PP/ST author should specify the [*set of users and/ or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies ‘a single user or subject’, the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.**

Selection:

678 **In FPR_PSE.1.1 the PP/ST author should specify whether authorised administrators (responsible for other classes) are included or excluded from the scope. If the PP/ST author specifies 'included' the set of subjects/users may include authorised administrators, but not the authorised administrator responsible for privacy.**

Assignment:

679 **In FPR_PSE.1.1 the PP/ST author should identify the [*list of subjects and/or operations*] where the users identity of the subject should be protected, for example 'the accessing of job offers'.**

FPR_PSE.2 Reversible Pseudonymity

User Application Notes

680 In this component the TSF shall ensure that under specified conditions the user identity related to a provided reference can be determined.

681 In FPR_PSE.1 the TSF shall provide an alias instead of the user identity. When the specified conditions are satisfied, the user identity to which the alias belong can be determined. An example of such a condition in an electronic cash environment is: "The TSF shall provide the notary a capability to determine the user identity based on the provided alias only under the conditions that a check has been issued twice.".

Operations

Assignment:

682 In FPR_PSE.2.1 the PP/ST author should specify the [*set of users and/or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies 'a single user or subject', the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.

Selection:

683 In FPR_PSE.2.1 the PP/ST author should specify whether authorised administrators (responsible for other classes) are included or excluded from the scope. If the PP/ST author specifies 'included' the set of subjects/users may include authorised administrators, but not the authorised administrator responsible for privacy.

Assignment:

684 In FPR_PSE.2.1 the PP/ST author should identify the [*list of subjects and/or operations*] where the users identity of the subject should be protected, for example 'the accessing of job offers'.

Selection:

685 **In FPR_PSE.2.3 the PP/ST author should select whether the authorised administrator and/or specific subjects can determine the user identity.**

Assignment:

686 **In FPR_PSE.2.3 the PP/ST author should identify the *list of trusted subjects* which can obtain the users identity under a specified condition, for example a notary or special administrative role.**

Assignment:

687 **In FPR_PSE.2.3 the PP/ST author should identify the [*list of conditions*] under which the subjects and authorised administrator can determine the users identity based on the provided reference. These conditions can be conditions such as time of day, or they can be administrative such as on a court order.**

FPR_PSE.3 Alias Pseudonymity

User Application Notes

688 In this component the TSF shall ensure that the provided reference meets certain construction rules and thereby can be used in a secure way by potentially insecure subjects.

689 If a user wants to use disk resources without disclosing its identity, pseudonymity can be used. However, every time the user accesses the system, the same alias must be used. These kind of conditions can be specified in this component.

Operations

Assignment:

690 In FPR_PSE.3.1 the PP/ST author should specify the [*set of users and/or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies ‘a single user or subject’, the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.

Selection:

691 In FPR_PSE.3.1 the PP/ST author should specify whether authorised administrators (responsible for other classes) are included or excluded from the scope. If the PP/ST author specifies ‘included’ the set of subjects/users may include authorised administrators, but not the authorised administrator responsible for privacy.

Assignment:

692 In FPR_PSE.3.1 the PP/ST author should identify the [*list of subjects and/ or operations*] where the users identity of the subject should be protected, for example ‘the accessing of job offers’.

Assignment:

693 **In FPR_PSE.3.3 the PP/ST author should identify the [*list of conditions*] which indicate when the used reference for the user-identity shall be identical and when it shall be different, for example “when the user logs on to the same host” it will use a unique alias.**

FPR_UNL Unlinkability

694 Unlinkability ensures that an entity may make multiple uses of resources or services without others being able to link these uses together. Unlinkability differs from pseudonymity that, although in pseudonymity the user is also not known, relations between different actions can be provided.

User notes

695 The requirements for unlinkability are intended to protect the user identity against the use of profiling of the operations. For example in case a telephone smart card is employed with a unique number, the telephone company can determine the behaviour of the user of this telephone card. When furthermore a telephone profile of the users is known, the card can be linked to a specific user. Hiding the relationship between different invocations of a service or access of a resource will prevent this kind of information gathering.

696 As a result, a requirement for unlinkability implies that the subject and user identity of an operation must be protected. Otherwise this information can be used to link operations together.

697 Unlinkability requires that different operations cannot be related. This relationship can take several forms. For example the user associated with the operation, or the terminal which initiated the action, or the time the action was executed. The PP/ST author can specify what kind of relationships are present which must be countered.

698 Possible applications include the ability to make multiple use of a pseudonym without creating a usage pattern that might disclose the user's identity.

699 Examples for potential hostile subjects and users are providers, system operators, communication partners and users, who smuggle malicious parts, (e.g. Trojan Horses) into systems, they do not operate but want to get information about. All of these attackers can investigate (e.g. which users used which services) and misuse this information. Unlinkability protects users from linkages, which could be drawn between several actions of a customer. An example is a series of phone calls made by an anonymous customer to different partners, where the combination of the partner's identities might disclose the identity of the customer.

FPR_UNL.1 Unlinkability

User Application Notes

700 This component ensures that users cannot link different operations in the system and thereby obtain information.

Operations

Assignment:

701 **In FPR_UNL.1.1 the PP/ST author should specify the [*set of users and/ or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies 'a single user or subject', the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.**

Selection:

702 **In FPR_UNL.1.1 the PP/ST author should specify whether authorised administrators (responsible for other classes) are included or excluded from the scope. If the PP/ST author specifies 'included' the set of subjects/users may include authorised administrators, but not the authorised administrator responsible for privacy.**

Assignment:

703 **In FPR_UNL.1.1 the PP/ST author should identify the [*list of operations*] which should be subjected to the unlinkability requirement, for example "sending email".**

Selection:

704 **In FPR_UNL.1.1 the PP/ST author should select which relationships should be obscured. The selection allows either the user identity or an assignment of relations to be specified.**

Assignment:

705 **In FPR_UNL.1.1 the PP/ST author might need to identify the [*list of relations*] which should be protected against, for example "originate from the same terminal".**

FPR_UNO Unobservability

706 Unobservability ensures that a user may use a resource or service without others, especially third parties, being able to observe that the resource or service is being used.

User notes

707 Unobservability approaches the user identity from a different direction than the previous families Anonymity, Pseudonymity, and Unlinkability. Instead of not releasing the users identity the fact that somebody is using the resource / service is hidden.

708 Sometimes regular users are not allowed to see the use of a resource, but an authorised administrator must be allowed to see the use of the resource in order to perform his duties. In those cases the FPR_UNO.2 could be requested, which provides the capability for an authorised administrator to see the usage.

709 Examples of potential hostile users or subjects are malicious systems operators or users, who smuggle malicious parts, e.g. Trojan Horses into system. Several countries consider the protection of communications unobservability as essential for the protection of constitutional rights.

FPR_UNO.1 Unobservability

User Application Notes

710 This component ensures that the use of a function cannot be observed by unauthorised users. In addition to this component a PP/ST author might want to incorporate Covert Channel Analysis.

Operations

Assignment:

711 **In FPR_UNO.1.1 the PP/ST author should specify the [*set of users and/ or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies ‘a single user or subject’, the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.**

Selection:

712 **In FPR_UNO.1.1 the PP/ST author should specify whether authorised administrators (responsible for other classes) are included or excluded from the scope. If the PP/ST author specifies ‘included’ the set of subjects/users may include authorised administrators, but not the authorised administrator responsible for privacy.**

Assignment:

713 **For FPR_UNO.1.1 the PP/ST author should identify the [*list of operations*] which are subjected to the unobservability requirement. In other words the other user/subjects cannot observe the operations in the specified list on a covered object, for example reading and writing on the object.**

Assignment:

714 **For FPR_UNO.1.1 the PP/ST author should identify the [*list of objects*] which are covered by the unobservability requirement. An example could be a specific mail server or ftp site.**

FPR_UNO.2 Authorised Administrator Observability

User Application Notes

715 This component is used to specify that there will be an authorised administrator with the rights to view the resource utilisation. Without this component, this review is allowed, but not mandated.

Operations

Assignment:

716 In FPR_UNO.2.1 the PP/ST author should specify the [*set of users and/or subjects*] against which, if they are working together, the TSF must provide protection. For example if the PP/ST author specifies ‘a single user or subject’, the TSF must be protected against each individual user or subject but might have some weaknesses with respect to cooperating users.

Selection:

717 In FPR_UNO.2.1 the PP/ST author should specify whether authorised administrators (responsible for other classes) are included or excluded from the scope. If the PP/ST author specifies ‘included’ the set of subjects/users may include authorised administrators, but not the authorised administrator responsible for privacy.

Assignment:

718 For FPR_UNO.2.1 the PP/ST author should identify the [*list of operations*] which are subjected to the unobservability requirement. In other words the other user/subjects cannot observe the operations in the specified list on a covered object, for example reading and writing on the object.

Assignment:

719 For FPR_UNO.2.1 the PP/ST author should identify the [*list of objects*] which are covered by the unobservability requirement. An example could be a specific mail server or ftp site

Class FPT

Protection of the Trusted Security Functions

720 This class contains families of functional requirements that relate to the integrity and management of the mechanisms that provide the TSF (independent of TSP-specifics) and to the integrity and management of TSF data (independent of the specific contents of the TSP data). In some sense, families in this class may appear to duplicate components in the FDP (User Data Protection) class, they may even be implemented using the same mechanisms; however, FDP focuses on user data protection, while FPT focuses on TSF data protection. In fact, components from the FPT class are necessary even in the absence of any user data protection, to provide confidence in the enforcement of other policies (such as accountability) that may be specified in the PP/ST.

721 From the point of view of this class, there are three significant portions that make up the TSF:

a) The TSF's *abstract machine*, which is the virtual or physical machine upon which the specific TSF software under evaluation executes.

b) The TSF's *software*, which executes on the abstract machine and implements the mechanisms that enforce the TSP.

c) The TSF's *data*, which are the administrative databases that guide the enforcement of the TSP.

722 All of the families in the FPT class can be related to these three areas, and fall into the following groupings:

a) Families that address protection of the TSF mechanism. These families are:

1) FPT_PHP (TSF Physical Protection) and FPT_SWM (TSF Software Modification), which provide authorised administrators with the ability to detect external attacks on the parts of the TOE that comprise the TSF.

2) FPT_AMT (Underlying Abstract Machine Test) and FPT_TST (TSF Self Test), which provide authorised administrators with the ability to verify correct operation of the TSF and integrity of the TSF data and underlying abstract machine.

3) FPT_SEP (Domain Separation) and FPT_RVM (Reference Mediation), which protect the TSF during execution and assure that the TSF cannot be bypassed. When appropriate components from these families are combined with the appropriate components from ADV_INT (TSF internals), the TOE can be said to have what has been traditionally called a “Reference Monitor.” The Reference Monitor is that portion of the TSF responsible for the enforcement of the TSP; it has the following three characteristics:

•Untrusted subjects cannot interfere with its operation; i.e., it is tamperproof. This is addressed by the components in the FPT_SEP family.

•Untrusted subjects cannot bypass its checks; i.e., it is always invoked. This is addressed by the components in the FPT_RVM family.

•It is simple enough to be analysed; i.e., it's design is conceptually simple. This is addressed by the components in the ADV_INT family.

4) FPT_RCV (Trusted Recovery), FPT_FLS Fail Secure, and FPT_TRC (Internal TOE TSF Data Replication Consistency), which address the behaviour of the TSF when failure occurs and immediately after.

5) FPT_ITA (Inter-TSF Availability of TSF Data), FPT_ITC (Inter-TSF Confidentiality of TSF Data), FPT_ITI (Inter-TSF Integrity of TSF Data), which address the protection and availability of TSF data between the TSF and a remote TSF. FPT_ITT (Internal TOE Transfer) is similar to the previous three families, but addresses protection of TSF data when it is transmitted between parts of the TOE.

6) FPT_RPL (Replay Detection and Prevention), which addresses the replay of various types of entities.

7) FPT_SSP State Synchrony Protocol, which addresses the state synchrony required between two TSF components.

8) FPT_STM (Time Stamps), which addresses timing consistency internal to the TSF.

b) Families that address the management of TSF data. These families are:

1) FPT_SAE Security Attribute Expiration, which addresses the expiration of the validity of security attributes.

2) FPT_REV Revocation, which address the revocation of security attributes.

3) FPT_TDC Inter-TSF TSF Data Consistency, which addresses the consistency of TSF data shared between TSF of distinct TOEs.

4) FPT_TSA (TOE Security Administration), which addresses the functions that must be available to the administrator that are independent of those related to any other class.

5) FPT_TSM (TOE Security Management), which addresses how management of the TSF is structured.

6) FPT_TSU (TOE Administrative Safe Use), which addresses the ease of use of the administrative interface.

FPT_AMT Underlying Abstract Machine Test

723 This family defines the requirements for the TSF's testing of security assumptions made about the underlying abstract machine upon which the TSF relies. This "abstract" machine could be a hardware/firware platform, or it could be some known and assessed hardware/software combination acting as a virtual machine. Examples could be testing hardware page protection, sending sample packets across a network to ensure receipt, verifying the behaviour of the virtual machine interface, etc. These tests can be carried out either in some maintenance state, at start-up, on-line, or continuously. The actions to be taken by the TOE as the result of self testing are defined in FPT_RCV.

User notes

724 The term "underlying abstract machine" typically refers to the hardware components upon which the TSF software functions have been implemented. However, the phrase can also be used to refer to an underlying, previously evaluated hardware and software combination behaving as a virtual machine.

724 The tests of the abstract machine may take various forms:

a) **Power-On Tests**. These are tests that ensure the correct operation of the underlying platform. For hardware and firmware, this might include tests of elements such as memory boards, data paths, buses, control logic, processor registers, communication ports, console interfaces, speakers, and peripherals. For software elements (virtual machine), this would include verification of correct initialisation and behaviour.

b) **Loadable Tests**. These are tests that might be loaded and executed by the administrator. This might include processor component stress tests (logic units, calculation units, etc.) and control memory.

Evaluator notes

725 The tests of the underlying abstract machine should be sufficient to test all of the characteristics of the underlying abstract machine upon which the TSF relies.

Documentation notes

726 The indicated assurance documentation (if applicable) shall contain the indicated information:

a) A description of the product features that can be used to periodically demonstrate the correct operation of the underlying abstract machine [AGD_ADM Administrator guidance].

b) A description of the coverage and use of the underlying abstract machine tests. [AGD_ADM Administrator guidance]

FPT_AMT.1 Periodic Abstract Machine Testing

User Application Notes

727 This component provides support for the periodic testing of the critical functions of the underlying abstract machine upon which the TSF's operation depends by requiring the ability to periodically invoke testing functions.

Evaluator application notes

728 It is acceptable for the functions that are available to the administrator for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access, during maintenance, to authorised administrators.

FPT_AMT.2 Abstract Machine Testing During Start-Up

User Application Notes

729 This component adds to the support for the periodic testing of the critical functions of the underlying abstract machine upon which the TSF's operation depends by calling for not only the ability to periodically invoke the tests, but to have tests executed as part of the TSF start-up mechanism.

Evaluator application notes

730 It is acceptable for the functions that are available to the administrator for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access, during maintenance, to authorised administrators.

FPT_AMT.3 Abstract Machine Testing During Normal Operation

User Application Notes

731 This component adds to the support for the periodic testing of the critical functions of the underlying abstract machine upon which the TSF's operation depends by calling for not only the ability to periodically invoke the tests and have the tests executed as part of the TSF start-up mechanism, but to have the TSF periodically perform the tests during normal operation.

Evaluator application notes

732 It is acceptable for the functions that are available to the administrator for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access, during maintenance, to authorised administrators.

FPT_FLS Fail Secure

733 The requirements of this family ensure that the TOE will not violate its TSP in the event of certain types of failures in the TSF.

Documentation notes

734 The indicated assurance documentation (if applicable) shall contain the indicated information:

a) The failure situations in which secure state is preserved, and any guidance to the administrator regarding additional recovery necessary [AGD_ADM Administrator guidance]

b) Identification of what constitutes a secure state [ADV_FSP Functional specification]

FPT_FLS.1 Failure with Preservation of Secure State

User Application Notes

735 The term “secure state” refers to a state in which the TSF data are consistent and the TSF continues correct enforcement of the TSP.

Operations

Assignment:

736 **For FPT_FLS.1.1, the PP/ST author should list those *types of failures* for which the TSF must “fail secure,” that is, must preserve a secure state and continue to correctly enforce the TSP.**

FPT_ITA Inter-TSF Availability of TSF Data

737 This family defines the rules for the prevention of loss of availability of TSF data moving between the TOE's TSF and other TSFs. This data could be TSF critical data such as passwords or keys or it could be TSF executable code.

User Application Notes

738 This family is used in a distributed system context where the TSF is providing TSF data to a remote TSF.

FPT_ITA.1 Inter-TSF Availability Within a Defined Availability Factor

Operations

Assignment:

739 **For FPT_ITA.1.1, the PP/ST author should specify the types of TSF data that are subject to the availability metric.**

Refinement:

740 **For FPT_ITA.1.1, the PP/ST should quantify the availability metric for the applicable TSF data.**

FPT_ITC Inter-TSF Confidentiality of TSF Data

741 This family defines the rules for the protection from unauthorised disclosure of TSF data moving between the TOE's TSF and another TSF. This data could be TSF critical data such as passwords, keys, audit records, or TSF executable code.

User Application Notes

742 This family is used in a distributed system context where the TSF is providing TSF data to a remote TSF.

FPT_ITC.1 Inter-TSF Confidentiality During Transmission

Evaluator application notes

743 With the technology available at the time of writing of the CC, the only practical means of satisfying this requirement involves either physical protection of the transmission lines, or the use of cryptographic functions.

FPT_ITI Inter-TSF Integrity of TSF Data

744 This family defines the rules for the protection from unauthorised and undetectable modification of TSF data moving between the TOE's TSF and other TSFs. This data could, for example, be TSF critical data such as passwords, keys, audit data, or TSF executable code.

User notes

745 This family is used in a distributed system context where the TOE's TSF is providing TSF data to a remote TSF. Note that a requirement that addresses modification detection or recovery upon receipt cannot be specified, as the mechanisms that a remote TSF will use to protect its data cannot be determined in advance.

Documentation notes

746 The indicated assurance documentation (if applicable) shall contain the indicated information:

a) Description of the means by which a remote TSF can detect modification of TSF data. [ADV_HLD High-level design]

FPT_ITI.1 Inter-TSF Detection of Modification

User Application Notes

747 This component should be used in situations where it is sufficient to detect when data have been modified. An example of such a situation is one in which the remote TSF can request the TOE's TSF to retransmit data when modification has been detected.

748 Note that the functions that detect modification may not be present on the local TSF; rather, the TSF should transmit the data in such a way that a remote TSF could detect modification if the remote TSF implemented the appropriate algorithm.

749 The desired strength of modification detection is a function of the algorithm used, ranging from weak checksumming and parity mechanisms that may fail to detect multiple bit changes, to more complicated cryptographic checksum approaches.

Evaluator application notes

750 With the technology available at the time of writing of the Common Criteria, the only practical means of satisfying this requirement involves either physical protection of the transmission lines, or the use of cryptographic functions.

Operations

Refinement:

751 **For FPT_ITI.1.1, the PP/ST should specify the *modification metric* to be used by the TSF.**

FPT_ITI.2 Inter-TSF Detection and Prevention of Modification

User Application Notes

752 This component should be used in situations where it is necessary to detect, prevent, or correct modifications of TSF critical data.

753 Note that the functions that detect or recover from modification may not be present on the local TSF; rather, the TSF should transmit the data in such a way that the functions of a remote TSF could detect or recover from modification if the remote TSF implemented the appropriate mechanism.

754 The desired strength of modification detection is a function of the mechanism used, ranging from weak checksumming and parity mechanisms that may fail to detect multiple bit changes, to more complicated cryptographic checksum approaches.

755 The approach taken to correct modification might be done through some form of error correcting checksum.

Evaluator application notes

756 With the technology available at the time of writing of the Common Criteria, the only practical means of satisfying this requirement involves the use of cryptographic functions or some form of checksum.

Documentation notes

757 In addition to the documentation required for all components in this family, the indicated assurance documentation (if applicable) shall contain the indicated information:

a) Description of the means by which a remote TSF can detect recover from modification of TSF data. [ADV_HLD High-level design]

Operations

Refinement:

758 **For FPT_ITI.2.1, the PP/ST should specify the desired strength of modification detection.**

Assignment:

759 **For FPT_ITI.2.2, the PP/ST author must define the *types of modification* from which the TSF must be capable of recovering.**

FPT_ITT Internal TOE Transfer

760 This family provides requirements that address protection of TSF data when it is transferred between parts of a TOE across an internal channel.

User notes

761 The determination of the degree of physical separation above which this family should apply depends on the intended environment of use. In a hostile environment, there may be risks arising from transfers between parts of the TOE separated by only a system bus. In more benign environments, the transfers may be across more traditional network media.

762 The refinement of “an approved method” allows a PP/ST author to specify a particular approach to the confidentiality or integrity protection, for example, physically-secured lines or cryptographic solutions.

Evaluator notes

763 Based on technology available at the time of the development of this document, the only practical mechanism available to a TSF to provide this protection is cryptographically-based.

FPT_ITT.1 Basic Internal TSF Data Transfer Protection

Operations

Assignment:

764 **In FPT_ITT.1.1, the PP/ST author should identify the *specific mechanism* to be used to provide the above identified type(s) of protection.**

Selection:

765 **In FPT_ITT.1.1, the PP/ST author should specify the desired type of protection to be provided from the choices: *disclosure, modification, disclosure and modification*.**

FPT_ITT.2 TSF Data Transmission Separation by Attribute

User Application Notes

766 One of the ways to achieve separation of channels based on SFP-relevant attributes is through the use of distinct encryption algorithms.

Operations

Assignment:

767 **In FPT_ITT.2.1, the PP/ST author should identify the *specific mechanism* to be used to provide the above identified type(s) of protection.**

Selection:

768 **In FPT_ITT.2.1, the PP/ST author should specify the desired type of protection to be provided from the choices: *disclosure, modification, disclosure and modification*.**

FPT_ITT.3 TSF Data Integrity Monitoring

Operations

Assignment:

769 **In FPT_ITT.3.1, the PP/ST author should identify the *specific mechanism* to be used to provide the above identified type(s) of protection.**

Selection:

770 **In FPT_ITT.3.1, the PP/ST author should specify the desired type of protection to be provided from the choices: *disclosure, modification, disclosure and modification*.**

Selection:

771 **In FPT_ITT.3.1, the PP/ST author must select the types of integrity errors that the TSF must be capable of detection. These integrity errors may include: *modification of data, substitution of data, re-ordering of data, deletion of data, or other integrity errors.***

Selection:

772 **In FPT_ITT.3.1, the PP/ST author should specify the desired type of modification that the TSF shall be able to detect. The PP/ST author should select from: *modification of data, substitution of data, re-ordering of data, and/or deletion of data*.**

Assignment:

773 **In FPT_ITT.3.3, the PP/ST author should specify any *other integrity errors* that the TSF must be capable of detecting.**

Assignment:

774 **In FPT_ITT.3.4, the PP/ST author should *specify the action to be taken* when an integrity error is identified.**

FPT_PHP TSF Physical Protection

775 TSF physical protection components refer to restrictions on unauthorised physical access to the TSF, and to the deterrence of, and resistance to, unauthorised physical use, modification, or substitution of the TSF.

776 The requirements of components in this family ensure that the TSF is either protected from physical tampering and interference, or operates in a protected environment. Satisfying the requirements of these components results in the TSF being packaged and used in such a manner that physical tampering is detectable, or resistance to physical tampering is measurable based on defined work factors. Without these components, the protection functions of a TSF lose their effectiveness in environments where physical damage cannot be prevented. This component also provides requirements regarding how the TSF must respond to physical tampering attempts.

User notes

777 Evaluators should review the administrator guidance and design documentation to ensure that all distinct devices and elements within the TSF's perimeter are described.

778 It is acceptable for the functions that are available to the administrator for detecting physical attack to be available only in an off-line or maintenance mode. Controls should be in place to limit access during such modes to authorised administrators. As the TSF may not be "operational" during those modes, it may be able to provide normal enforcement for administrator access.

Documentation notes

779 The indicated assurance documentation (if applicable) shall contain the following information:

a) Identification of the TSF's physical perimeter. [AGD_ADM Administrator guidance]

b) The distinct devices and elements within the TSF's physical perimeter. [AGD_ADM Administrator guidance]

FPT_PHP.1 Passive Detection of Physical Attack

User Application Notes

780 Component FPT_PHP.1 should be used when threats from unauthorised physical tampering with parts of the TOE are not countered by procedural methods. It addresses the threat of undetected actual physical tampering with the TSF.

FPT_PHP.2 Notification of Physical Attack

User Application Notes

781 Component FPT_PHP.2 should be used when threats from unauthorised physical tampering with parts of the TOE are not countered by procedural methods, and it is required that designated individuals be notified of physical attacks. It addresses the threat that physical tampering with TSF elements, although detected, may not be noticed.

Operations

Assignment:

782 **For FPT_PHP.2.3, the PP/ST author must provide a *list of devices/ elements for which active detection* of physical tampering *is required*.**

Assignment:

783 **For FPT_PHP.2.3, the PP/ST author must identify the type of administrative *user or role* that is to be notified when tampering is detected. The administrative user or role may vary depending on the particular security administration component (from the FPT_TSA family) included in the PP/ST.**

FPT_PHP.3 Resistance to Physical Attack

784 For some forms of attack, it is necessary that the TOE not only detects the attack, but actually resists the attack or delays the attacker.

User Application Notes

785 This component should be used when TSF devices and elements are expected to operate in an environment where observation, analysis, or modification of the internals of a TSF device or element itself is a threat. This component partially addresses the threat of the TSF violating the TSP as the result of an actual physical attack, by providing increased resistance to attack.

Evaluator application notes

786 The determination of acceptable work factors is by its very nature somewhat qualitative, and cannot always be evaluated in a reasonable time or in a repeatable fashion. Evaluator judgement will be required to determine if a particular attack scenario resistance mechanism would require the indicated level of effort.

Operations

Assignment:

787 **For FPT_PHP.3.3, the PP/ST author must provide a *list of devices/ elements for which active detection* of physical tampering *is required*.**

Assignment:

788 **For FPT_PHP.3.3, the PP/ST author must identify the type of administrative *user or role* that is to be notified when tampering is detected. The administrative user or role may vary depending on the particular security administration component (from the FPT_TSA family) included in the PP/ST.**

Assignment:

789 **For FPT_PHP.3.4, the PP/ST author must specify both the devices/ elements for which the TSF must resist physical tampering attacks, and the specific attack scenario that must be countered. This list may be applied to a defined subset of the TSF physical devices and elements based on considerations such as technology limitations and relative physical exposure of the device. Such subsetting must be clearly defined and justified.**

790 **The specific TSP(s) of relevance may be listed in FPT_PHP.3.4.**

791 **FPT_PHP.3.4 may include specific acceptance criteria for the work factor parameters.**

Assignment:

792 **For FPT_PHP.3.5, the PP/ST author must specify both the devices/ elements for which the TSF must automatically respond to physical tampering attacks, and the specific attack scenarios that must be countered. This list may be applied to a defined subset of the TSF physical devices and elements based on considerations such as technology limitations and relative physical exposure of the device. Such subsetting must be clearly defined and justified. The automatic response must be such that the policy of the device is preserved; for example, with a confidentiality policy, it would be acceptable to physically disable the device to that the protected information may not be retrieved.**

FPT_RCV Trusted Recovery

793 The requirements of this family ensure that the TSF can determine that the TOE is started-up without protection compromise and can recover without protection compromise after discontinuity of operations. Satisfying the requirements of this family establishes that the initial and recovered states of the TSF satisfy the requirements. This family is important because the start-up state of the TSF determines the protection of subsequent states.

794 Recovery components reconstruct the TSF secure states or prevent transitions to insecure states as a direct response to occurrences of expected failures, discontinuity of operation or start-up. Failures that must be generally anticipated include the following:

a) Unmaskable action failures that always result in a system crash (e.g., persistent inconsistency of critical system tables, uncontrolled transfers within the TSF code caused by transient failures of hardware or firmware, power failures, processor failures, communication failures).

b) Media failures causing part or all of the media representing the TSF objects to become inaccessible or corrupt (e.g., parity errors, disk head crash, persistent read/write failure caused by misaligned disk heads, worn-out magnetic coating, dust on the disk surface).

c) Discontinuity of operation caused by erroneous administrative action or lack of timely administrative action (e.g., unexpected shutdowns by turning off power, ignoring the exhaustion of critical resources, inadequate installed configuration).

795 Note that recovery may be from either a complete or partial failure scenario. Although a complete failure might occur in a monolithic operating system, it is less likely to occur in a distributed environment. In such environments, subsystems may fail, but other portions remain operational. Further, critical components may be redundant (disk mirroring, alternative routes), and checkpoints may be available. Thus, recovery is expressed in terms of recovery to some previously known secure state.

796 Mechanisms designed to detect exceptional conditions during operation fall under FPT_SWM TSF Software Modification, FPT_TST TSF Self Test, FPT_FLS Fail Secure, and other areas that address the concept of “Software Safety.”

User notes

797 Throughout this family, the phrase “secure state” is used. This refers to some state in which the TOE is known to have consistent TSF data and a TSF that can correctly enforce the policy. This state may be the initial “boot” of a clean system, or it might be some checkpointed state.

Documentation notes

798 The indicated assurance documentation (if applicable) should contain the following information:

a) Identification of what constitutes a secure state [ADV_FSP Functional specification]

b) Identification of the failures and service discontinuities for which (1) recovery is possible, and (2) recovery is not possible and reinstallation of the TSF is required [ADV_HLD High-level design]

c) Manual procedures to be followed to recover to a secure state. [AGD_ADM Administrator guidance]

FPT_RCV.1 Manual Recovery

799 In the hierarchy of trusted recovery, recovery that requires only manual intervention is the least desirable, for it precludes the use of the system in an unattended fashion.

User Application Notes

800 This component is intended for use in TOEs that do not require unattended recovery to a secure state. The requirements of this component reduce the threat of protection compromise resulting from an attended TOE returning to an insecure state after recovery from a failure or other discontinuity.

Evaluator application notes

801 It is acceptable for the functions that are available to the administrator for trusted recovery to be available only in a maintenance mode. Controls should be in place to limit access during maintenance to authorised administrators.

FPT_RCV.2 Automated Recovery

802 Automated recovery is considered to be more useful than manual recovery, as it allows the machine to operate in an unattended fashion.

User Application Notes

803 The component FPT_RCV.2 extends the feature coverage of FPT_RCV.1 by requiring that there be at least one automated method of recovery from failure or service discontinuity. It addresses the threat of protection compromise resulting from an unattended TOE returning to an insecure state after recovery from a failure or other discontinuity.

Evaluator application notes

804 It is acceptable for the functions that are available to the administrator for trusted recovery to be available only in a maintenance mode. Controls should be in place to limit access during maintenance to authorised administrators.

805 For FPT_RCV.2.1, it is the responsibility of the developer of the TSF to determine the set of recoverable failures and service discontinuities.

806 It is assumed that the evaluators will verify the robustness of the automated recovery mechanisms.

Operations

Assignment:

807 **For FPT_RCV.2.3, the PP/ST author must specify the *list of failures or other discontinuities* for which automated recovery shall be possible.**

FPT_RCV.3 Automated Recovery without Undue Loss

808 Automated recovery is considered to be more useful than manual recovery, but it runs the risk of losing a substantial number of objects. Preventing undue loss of objects provides additional utility to the recovery effort.

User Application Notes

809 The component FPT_RCV.3 extends the feature coverage of FPT_RCV.2 by requiring that there not be undue loss of TSF data or objects within the TSC. At FPT_RCV.2, the automated recovery mechanisms could conceivably recover by deleting all objects and returning the TSF to a known secure state. This type of drastic automated recovery is precluded in FPT_RCV.3.

810 This component addresses the threat of protection compromise resulting from an unattended TOE returning to an insecure state after recovery from a failure or other discontinuity with a large loss of TSF data or objects within the TSC.

Evaluator application notes

811 It is acceptable for the functions that are available to the administrator for trusted recovery to be available only in a maintenance mode. Controls should be in place to limit access during maintenance to authorised administrators.

812 It is assumed that the evaluators will verify the robustness of the automated recovery mechanisms.

Documentation notes

813 In addition to the documentation requirements applicable to all components in this family, the following assurance documentation, if applicable, shall contain the indicated information:

a) The approach taken to determine the objects that were or were not capable of being recovered. [ADV_HLD High-level design]

Operations

Assignment:

814 **For FPT_RCV.3.3, the PP/ST author must specify the *list of failures or other discontinuities* for which automated recovery shall be possible.**

Assignment:

815 **For FPT_RCV.3.4, the PP/ST author should provide a *quantification* for the amount of loss of TSF data or objects that is acceptable.**

FPT_RCV.4 Function Recovery

815 For selected SFs in the TSF, it is necessary that the SF fail in a manner that does not result in compromised TSF data.

Operations

Assignment:

816 **In FPT_RCV.4.1, the PP/ST author should *list the SFs and failure scenarios* for which the TSF should return to its state immediately before SF invocation.**

FPT_REV Revocation

817 This family addresses revocation of security attributes for a variety of entities within a TOE.

Documentation notes

818 AGD_ADM Administrator Guidance must describe the timing aspects of the revocation. This is especially important for TSF's with distributed architecture.

FPT_REV.1 Basic Revocation

Operations

Selection:

819 **In FPT_REV.1.1, the PP/ST author should specify whether the ability to revoke attributes from *users, subjects, and/or objects* shall be provided by the TSF.**

Assignment:

820 **In FPT_REV.1, the PP/ST author must specify *the categories of additional resources* within the TSC for which a capability to revoke attributes is required.**

Assignment:

821 **In FPT_REV.1.2, the PP/ST author must provide a *specification of revocation rules*. Examples of this specification could include: prior to the next operation on the associated resource, or for all new subject creations.**

FPT_REV.2 Immediate Revocation

User Application Notes

822 In distributed systems, immediate revocation may not be possible due to unanticipated domain disconnections. Refinement of the word "immediate" may be necessary in those cases. A potential refinement is that all current access rights should be re-evaluated with the new set of security attributes when the TSF mediating that access becomes aware of the revocation.

Operations

Selection:

823 **In FPT_REV.1.1, the PP/ST author should specify whether the ability to revoke attributes from *users, subjects, and/or objects* shall be provided by the TSF.**

Assignment:

824 **In FPT_REV.1, the PP/ST author must specify *the categories of additional resources* within the TSC for which a capability to revoke attributes is required.**

FPT_RPL Replay Detection and Prevention

825 This family addresses prevention of replay for various types of entities.

FPT_RPL.1 Replay Detection and Prevention

User Application Notes

826 The entities included here are, for example, messages, service requests, service responses, or sessions.

Operations

Assignment:

827 **In FPT_RPL.1.1, the PP/ST author must provide a *list of identified entities* for which detection of replay must be possible. Examples of such entities might include: messages, service requests, service responses, and user sessions.**

828 **In FPT_RPL.1.2, the PP/ST author must specify the *list of actions* to be taken by the TSF when replay is detected. The potential set of actions that can be taken includes: ignoring the replayed entity, requesting confirmation of the entity from the identified source, terminating the subject from which the re-played entity originated.**

FPT_RVM Reference Mediation

829 The components of this family address the “always invoked” aspect of a traditional reference monitor. The goal of these components is to ensure, with respect to a given SFP, that all actions requiring policy enforcement invoked by subjects untrusted with respect to any or all of that SFP to objects controlled by that SFP subjects are validated by the TSF against the SFP. If the portion of the TSF that enforces the SFP also meets the requirements of appropriate components from FPT_SEP (Domain Separation) and ADV_INT (TSF internals), than that portion of the TSF provides a “reference monitor” for that SFP.

830 A TSF that implements a SFP provides effective protection against unauthorised operation if and only if all enforceable actions (e.g., accesses to objects) issued by untrusted with respect to any or all of that SFP subjects are validated by the TSF before succeeding. If the enforceable action is incorrectly enforced or bypassed, the overall enforcement of the SFP has been compromised. “Untrusted” subjects could then bypass the SFP in a variety of unauthorised ways (e.g., circumvent access checks for some subjects or objects, bypass checks for objects whose protection was assumed by applications, retain access rights beyond their intended lifetime, bypass auditing of audited actions, or bypass authentication). Note that the term “untrusted subject” refers to subjects untrusted with respect to the any or all of specific SFPs being enforced; a subject may be trusted with respect to one SFP and untrusted with respect to a different SFP.

FPT_RVM.1 Non-Bypassability of the TSP

User Application Notes

831 In order to obtain the equivalent of a reference monitor, this component must be used with either FPT_SEP.2 (Reference Monitor for some SFPs) or FPT_SEP.3 (Complete Reference Monitor), and ADV_INT.3 (Minimisation of Complexity). Further, if complete reference mediation is required, the components from Class FDP must cover all objects.

FPT_SAE Security Attribute Expiration

832 This family addresses the capability to enforce time limits for the validity of security attributes. This family can be applied to specify expiration requirements for access control attributes, identification and authentication attributes, audit attributes, etc.

Documentation notes

833 AGD_ADM Administrator Guidance should list the attributes for which expiration times may be specified.

FPT_SAE.1 Time-Limited Authorisation

Operations

Assignment:

834 **For FPT_SAE.1.1, the PP/ST author must provide the *list of security attributes for which expiration is to be supported.* An example of such an attribute might be a user's security clearance.**

Assignment:

835 **For FPT_SAE.1.2, the PP/ST author must provide a *list of actions to be taken for each security attribute* that may have an expiration time.**

FPT_SEP Domain Separation

836 The components of this family ensure that at least one security domain is available for the TSF's own execution, and that the TSF is protected from external interference and tampering (e.g., by modification of TSF code or data structures) by untrusted subjects. Satisfying the requirements of this family makes the TSF self-protecting, meaning that an untrusted subject cannot modify or damage the TSF.

837 This family requires the following:

a) The resources of the TSF's security domain ("protected domain") and those of subjects and unconstrained entities external to the domain are separated such that the entities external to the protected domain cannot observe or modify data structures or code internal to the protected domain.

b) The transfers between domains are controlled such that arbitrary entry to, or return from, the protected domain is not possible.

c) The user or application parameters passed to the protected domain by addresses are validated with respect to the protected domain's address space, and those passed by value are validated with respect to the values expected by the protected domain.

d) The security domains of subjects are distinct except for controlled sharing via the TSF.

User notes

838 This family is needed whenever confidence is required that the TSF has not been subverted.

839 In order to obtain the equivalent of a reference monitor, the components FPT_SEP.2 (Reference Monitor for some SFPs) or FPT_SEP.3 (Complete Reference Monitor) from this family must be used in conjunction with FPT_RVM.1 (Non-Bypassability of the TSP), and ADV_INT.3 (Minimisation of Complexity). Further, if complete reference mediation is required, the components from Class FDP must cover all objects.

Evaluator notes

840 In all components, the term "address space" is used in the generic sense of the set of resources accessible by the subject.

Documentation notes

841 The indicated assurance documentation (if applicable) shall contain the indicated information:

a) Description of the architecture and design of the domain separation mechanism [ADV_HLD High-level design, ADV_LLD Low-level design]

FPT_SEP.1 TSF Domain Separation

842 Without a separate protected domain for the TSF, there can be no assurance that the TSF has not been subjected to any tampering attacks by untrusted subjects. Such attacks may involve modification of the TSF code and/or TSF data structures.

User Application Notes

843 This component does not imply the presence of a reference monitor, as there is no mandatory distinct reference monitor domain.

FPT_SEP.2 Reference Monitor for some SFPs

844 The most important function provided by a TSF is the enforcement of its SFPs. In order to ensure that those significant SFPs exhibit the characteristics of a reference monitor (RM), in particular, being tamperproof, they must be in a domain distinct from the remainder of the TSF.

User Application Notes

845 To have a complete reference monitor for the indicated SFPs, this component must be used in conjunction with component FPT_RVM.1 (Non-Bypassability of the TSP) and ADV_INT.3 (Minimisation of Complexity).

Evaluator application notes

846 It is possible that a reference monitor in a layered design may provide functions beyond those of the SFPs. This arises out of the practical nature of layered software design. The goal should be to minimise the non-SFP related functions.

847 Note that it is acceptable for the reference monitors for all included SFPs to be in a single distinct reference monitor domain, as well as having multiple reference monitor domains (each enforcing one or more SFPs). If multiple reference monitor domains for SFPs are present, it is acceptable for them to be either peers or hierarchical.

848 For FPT_SEP.2.1, the phrase “unisolated portion of the TSF” refers to that portion of the TSF consisting of those SFPs not covered by FPT_SEP.2.3 and the non-SFP functions.

Documentation notes

849 In addition to the documentation requirements applicable to all components in the FPT_SEP family, the following assurance documentation, if applicable, shall contain the indicated information:

a) A description of how the TSF satisfies the requirements of a reference monitor for the indicated SFPs. [ADV_HLD High-level design]

Operations

Assignment:

850 **For FPT_SEP.2.3, the PP/ST author shall specify *the access control and information flow SFPs* in the TSP that must have a separate domain.**

FPT_SEP.3 Complete Reference Monitor

851 The most important function provided by a TSF is the enforcement of its SFPs. In order to ensure that the TSF exhibits the characteristics of a reference monitor (RM), in particular, being tamperproof, each TSP must be enforced in a domain distinct from the remainder of the TSF.

User Application Notes

852 To have a complete reference monitor for the indicated SFPs, this component must be used in conjunction with component FPT_RVM.1 (Non-Bypassability of the TSP) and ADV_INT.3 (Minimisation of Complexity).

Evaluator application notes

853 It is possible that a reference monitor in a layered design may provide functions beyond those of the SFPs. This arises out of the practical nature of layered software design. The goal should be to minimise the non-SFP related functions.

854 Note that it is acceptable for the reference monitors for all included SFPs to be in a single distinct reference monitor domain, as well as having multiple reference monitor domains (each enforcing one or more SFPs). If multiple reference monitor domains for SFPs are present, it is acceptable for them to be either peers or hierarchical.

Documentation notes

855 In addition to the documentation requirements applicable to all components in the FPT_SEP family, the following assurance documentation, if applicable, shall contain the indicated information:

a) A description of how the TSF satisfies the requirements of a reference monitor for the TSP. [ADV_HLD High-level design]

FPT_SSP State Synchrony Protocol

855 A myriad of actions in distributed systems gain complexity over their mainframe equivalents for reasons such as message time delay and state synchrony revocation, permission, encryption key invocation, audit, and database update. In most cases synchronisation of state between distributed functions involves an exchange protocol, not a simple action. When malice exists in the distributed environment of these protocols, more complex defensive protocols are required.

855 FPT_SSP establishes the requirement for certain critical security functions of the TSF to use this trusted protocol. FPT_SSP ensures that two distributed parts of the TOE (e.g., hosts) have synchronised their states after a security-relevant action.

User notes

856 Some states may never be synchronised, or the transaction cost may be too high for practical use; encryption key revocation is an example, where knowing the state after the revocation action is initiated, can never be known. Either the action was taken and acknowledgment cannot be sent, or the message was ignored by hostile and the revocation never occurred. Indeterminacy is unique to distributed systems. Indeterminacy and state synchrony are related, and the same solution may apply. It is futile to design for indeterminate states; the PP/ST author should express other requirements in such cases (e.g., raise an alarm, audit the event).

FPT_SSP.1 Simple Trusted Acknowledgement

User Application Notes

857 In this component, the TSF must be able to supply an acknowledgement to another TSF when requested by that other TSF. This acknowledgement should indicate that the TSF successfully received an unmodified transmission from the remote TSF.

FPT_SSP.2 Mutual Trusted Acknowledgement

User Application Notes

858 In this component, in addition to being able to provide an acknowledgement for the receipt of a data transmission, the TSF must be able to comply with a remote TSF's request for an acknowledgement to an acknowledgement.

859 For example, the local TSF transmits some data to a remote TSF. The remote TSF acknowledges the successful receipt of the data and requests that the sending TSF confirm that it receives the acknowledgement. This mechanism provides additional confidence that both TSFs involved in the data transmission know that the transmission completed successfully.

FPT_STM Time Stamps

859 This family addresses requirements for a trusted time stamp function within a TOE.

User notes

860 It is the responsibility of the PP/ST author to clarify the meaning of the phrase “trusted time stamp”, and to indicate where the responsibility lies in determining the acceptance of trust.

FPT_STM.1 Trusted Time Stamps

Operation : No permitted operation.

FPT_SWM TSF Software Modification

861 The requirements of this family are needed to detect the corruption of TSF code and data structures by various failures that do not necessarily stop the TOE's operation (which would be handled by other families). These checks must be performed because these failures may not necessarily be prevented. Such failures can occur either because of unforeseen failure modes or associated oversights in the design of hardware, firmware, or software, or because of malicious corruption of the TSF due to inadequate logical and/or physical protection.

User notes

862 This component should be used whenever the integrity requirements of executables is significant.

FPT_SWM.1 Protection of Executables

Operation : No permitted operation.

FPT_TDC Inter-TSF TSF Data Consistency

862 In a distributed or composite system environment, a TOE may need to exchange TOE data (e.g., the SFP-attributes associated with data, audit information, identification information) with the TSF of a distinct TOE. This family defines the requirements for sharing and consistent interpretation of these attributes between the TSFs of different TOEs.

User notes

863 The components in this family are intended to provide requirements for automated support for TSF data consistency when it is transferred between distinct TOEs. It is also possible that wholly procedural means could be used to produce security attribute consistency, but they are not provided for here.

864 This family is different than FDP_ETC and FDP_ITC because those two families are concerned with resolving the security attributes between the TSF and its import/ export medium only.

865 If the integrity of the TSF data is of concern, requirements should be chosen from the FPT_ITI family. These components specify requirements for the TSF to be able to detect or detect and correct modifications to TSF data in transit.

866 FDP_SAC is concerned with resolving security attributes between the TSFs of two TOEs (the end points of the communication). In other words, FDP_SAC enforces the security attribute protocols and conventions between the TSFs. In addition, this protocol may also specify how to recognise unauthorised modification of the security attributes (whether malicious or not) and attempt to recover the original security attributes transmitted from the TSF at the other end. The integrity of the security attributes will be enforced by the dependencies.

Documentation notes

867 The ADV_FSP Functional specification assurance documentation should contain a specification of the manner in which the TSF maintains consistency of TSF data between the TSF of the TOEs, what attribute conventions are supported, and which possible TSF data modifications can potentially be corrected.

868 In a distributed systems environment, the AGD_ADM documentation should include a description of how the TOE would be used with other TOEs to form a network that provided enforcement of a consistent overall security policy. This description should also include any restrictions on interconnection, and all configuration information necessary to ensure proper operation.

FPT_TDC.1 Inter-TSF Basic TSF Data Consistency

User Application Notes

869 The TSF is responsible for maintaining the consistency of TSF data used by or associated with the specified function and that are common between two or more security domains. For example, the TSF data for the TSFs of two different TOEs

may have different conventions internally. For the TSF data to be used properly (e.g., to afford the user data the same protection as on the sending TSF) by the receiving TSF, the TSFs must use a pre-established protocol to exchange TSF data.

Operations

Assignment:

869 **In FPT_TDC.1.1, the PP/ST author must define the *list of TSF data* that shall be consistently interpreted when shared between TSFs.**

Assignment:

869 **In FPT_TDC.1.2, the PP/ST must assign the *list of rules, protocols, conventions, and standards for consistent communication*.**

FPT_TRC Internal TOE TSF Data Replication Consistency

870 The requirements of this family are needed to ensure the consistency of TSF data when such data is replicated internal to the TOE. Such data may become inconsistent if the internal channel between parts of the TOE becomes inoperative. If the TOE is internally structured as a network of parts of the TOE, this can occur when parts become disabled, network connections are broken, and so on.

User notes

871 The method of ensuring consistency is not specified in this component. It could be attained through a form of transaction logging, where appropriate transactions are "rolled back" to a site upon reconnection; it could be updating the replicated data through a synchronisation protocol. If a particular protocol is necessary for a PP/ST, it can be specified through refinement.

872 It may be impossible to synchronise some states, or the cost of such synchronisation may be too high. Examples of this situation are communication channel and encryption key revocations. Indeterminate states may also occur; if a specific behaviour is desired, it should be specified via refinement.

FPT_TRC.1 Internal TOE Data Consistency

Operations

Assignment:

873 **In FPT_TRC.1.2, the PP/ST author should *list the SFs that depend on TSF data replication consistency*.**

FPT_TSA TOE Security Administration

874 The TSF includes security administration families to allow authorised administrators to control the secure operation of the TOE, and to restrict the accessibility of security management functions to authorised users. At higher levels of this family, the administrative function is subdivided into distinct roles.

User notes

875 This family addresses the general administration of TSF administrative actions. The requirements for administration for each policy (be it user data protection, identification/authentication, or audit) are provided by families in each of the corresponding classes.

Documentation notes

876 The indicated assurance documentation (if applicable) shall contain the indicated information:

a) Specification of the security-relevant administrative functions in the TSF [ADV_FSP Functional specification]

b) Listing of the initial configuration of the security-relevant administrative commands and (if applicable) the roles with which they are associated [AGD_ADM Administrator guidance]

c) Information on the TSF facilities used by an authorised administrator to define security-relevant administrative commands and (if applicable) associate them with a role [AGD_ADM Administrator guidance]

d) A description of the responsibilities of the security-relevant administrative role(s), as applicable [AGD_ADM Administrator guidance]

FPT_TSA.1 Basic Security Administration

877 At a minimum, the TSF must provide the ability to associate security-relevant functions with specifically authorised users, or there can be no assurance that security management is controlled.

User Application Notes

878 This component addresses threats from users inappropriately invoking security-relevant administrative TSF functions. It should be used when minimal protection of security management functions is required. This minimal protection is achieved by explicitly associating security-relevant administrative functions with specific authorised users.

879 This component requires that information be maintained to identify whether a user is authorised to use a particular security-relevant administrative function.

Evaluator application notes

880 For FPT_TSA.1.1 and FPT_TSA.1.2, the security-relevant administrative functions must minimally include all functions necessary to securely manage a given TOE, even if those functions are not explicitly listed in the requirement. The TOE developer must define these functions and justify that they are all included.

881 For FPT_TSA.1.3, the TSF must provide some explicit mechanism to identify the users that are authorised to use security-relevant functions. A TOE that claims that all users are authorised to use security-relevant administrative functions is not acceptable.

882 It is acceptable for many of the security administration functions to be available only in a maintenance or off-line mode. FPT_TSA.1.3 applies only to those functions available through the TSF interface.

Operations

Assignment:

883 **In FPT_TSA.1.2, the PP/ST author should list any specific security-relevant administrative functions considered minimally acceptable for this requirement. These minimal functions will depend on the other components included in the PP/ST, but can include functions such as audit log maintenance, user authentication data management, and system initialisation and configuration.**

FPT_TSA.2 Separate Security Administrative Role

884 In situations where security administration is appropriate, it is also appropriate to ensure that users do not have access to administrative functions.

User Application Notes

885 This component is intended for use in TOEs that do not require any fine-grained controls other than a distinction between administrative and non-administrative users. There is no requirement for a non-security-relevant administrator role, although such a division is recommended. Actions taken by a non-security-relevant administrator are beyond the scope of this component.

886 This component addresses the threat of damage resulting from users performing administrative functions. It also addresses the threat that inadequate mechanisms have been provided to securely administer the TSF.

887 This component requires that information be maintained to identify whether a user is authorised to use a particular security-relevant administrative function.

Evaluator application notes

888 For FPT_TSA.2.1 and FPT_TSA.2.2, the security-relevant administrative functions must minimally include all functions necessary to securely manage a given TOE,

even if those functions are not explicitly listed in the requirement. The TOE developer must define and justify that all such functions are included.

889 For FPT_TSA.2.3, the TSF must provide some explicit mechanism to identify the users that are authorised to use security-relevant functions. A TOE that claims that all users are authorised to use security-relevant administrative functions is not acceptable.

890 It is acceptable for many of the security administration functions to be available only in a maintenance or off-line mode. FPT_TSA.2.3 applies only to those functions available through the TSF interface.

Operations

Assignment:

891 **In FPT_TSA.2.2, the PP/ST author should list any specific security-relevant administrative functions considered minimally acceptable for this requirement. These minimal functions will depend on the other components included in the PP/ST, but can include functions such as audit log maintenance, user authentication data management, and system initialisation and configuration.**

FPT_TSA.3 Multiple Security Administrative Roles

User Application Notes

892 This component should be used when fine-grained administrative roles are necessary. The nature of the roles may be as simple or as complex as is necessary. A simple definition might be distinguishing between routine operator functions (such as backup and restore) and other administration actions, such as audit analysis or account administration. Alternatively, it might be complex, with a distinct administrative user for each category of policy supported on the TSF; for example, account administrator, auditor, confidentiality-policy administrator.

893 In addition to addressing the threat of damage resulting from unprivileged users performing administrative functions, this component addresses the threat of authorised administrators abusing their authority by taking actions outside their assigned functional responsibilities.

894 This component requires that information be maintained to identify whether a user is authorised to use a particular security-relevant administrative function.

Evaluator application notes

895 For FPT_TSA.3.1 and FPT_TSA.3.2, the security-relevant administrative functions must minimally include all functions necessary to securely manage a given TOE, even if those functions are not explicitly listed in the requirement. The TOE developer must define these functions and justify that they are all included.

896 For FPT_TSA.3.7, it is anticipated that most security-relevant functions will be assigned to only one role. However, there may be selected common utilities that will be accessible to multiple roles (for example, electronic mail capabilities to notify users). Functions available to more than one role will require justification.

897 For FPT_TSA.3.3, the TSF must provide some explicit mechanism to identify the users that are authorised to use security-relevant functions. A TOE that claims that all users are authorised to use security-relevant administrative functions is not acceptable.

898 For FPT_TSA.3.3, it is acceptable for a security-relevant administrative function to be available in more than one role.

899 It is acceptable for many of the security administration functions to be available only in a maintenance or off-line mode. FPT_TSA.3.3 applies only to those functions available through the TSF interface.

900 For FPT_TSA.3.5, the TSF must provide some explicit mechanism to identify the users that are authorised to use security-relevant functions and the roles for which they are authorised. A TOE that claims that all users are authorised for all security-relevant administrative roles is not acceptable.

Documentation notes

901 In addition to the documentation requirements applicable to all components in this family, the following assurance documentation, if applicable, shall contain the indicated information:

a) Any administrative roles specifically designed into the TSF [ADV_FSP Functional specification, AGD_ADM Administrator guidance]

Operations

Assignment:

902 **In FPT_TSA.3.2, the PP/ST author should list any specific security-relevant administrative functions considered minimally acceptable for this requirement. These minimal functions will depend on the other components included in the PP/ST, but can include functions such as audit log maintenance, user authentication data management, and system initialisation and configuration.**

Assignment:

903 **In FPT_TSA.3.7, the PP/ST author should list all security-relevant administrator roles defined for the TOE. The roles selected will depend on other components included in the PP/ST, but can include roles such as operator, account administrator, auditor, TSP administrator, and system manager.**

FPT_TSA.4 Well-Defined Administrative Roles

User Application Notes

904 This component should be used when well-defined roles are required. Users assuming these roles are restricted by the TSF to only those functions required to perform those roles. As this restriction requires administrative roles to communicate only with the TSF, a dependency on Trusted Path is introduced.

905 This component reduces the likelihood of damage resulting from unprivileged users and from authorised administrators abusing their authority by taking actions outside their assigned functional responsibilities. It also addresses the threat that inadequate mechanisms have been provided to securely administer the TSF.

906 This component requires that information be maintained to identify whether a user is authorised to use a particular security-relevant administrative function.

Evaluator application notes

907 For FPT_TSA.4.1 and FPT_TSA.4.2, the security-relevant administrative functions must minimally include all functions necessary to securely manage a given TOE, even if those functions are not explicitly listed in the requirement. The TOE developer must define these functions and justify that they are all included.

908 For FPT_TSA.4.3, it is anticipated that most security-relevant functions will be assigned to only one role. However, there may be selected common utilities that will be accessible to multiple roles (for example, electronic mail capabilities to notify users). Functions available to more than one role will require justification.

909 For FPT_TSA.4.3, the TSF must provide some explicit mechanism to identify the users that are authorised to use security-relevant functions and the roles for which they are authorised. A TOE that claims that all users are authorised for all security-relevant administrative roles is not acceptable.

910 It is acceptable for many of the security administration functions to be available only in a maintenance or off-line mode. FPT_TSA.4.3 applies only to those functions available through the TSF interface.

Documentation notes

911 In addition to the documentation requirements applicable to all components in this family, the following assurance documentation, if applicable, shall contain the indicated information:

a) Any administrative roles specifically designed into the TSF [ADV_FSP Functional specification, AGD_ADM Administrator guidance]

Operations

Assignment:

912 **In FPT_TSA.4.2, the PP/ST author should list any specific security-relevant administrative functions considered minimally acceptable for this requirement. These minimal functions will depend on the other components included in the PP/ST, but can include functions such as audit log maintenance, user authentication data management, and system initialisation and configuration.**

Assignment:

913 **In FPT_TSA.4.3, the PP/ST author should list all security-relevant administrator roles defined for the TOE. The roles selected will depend on other components included in the PP/ST, but can include roles such as operator, account administrator, auditor, TSP administrator, and system manager.**

FPT_TSM TOE Security Management

914 The TSF of a TOE should provide security management functions to enable authorised administrators to set up and control the secure operation of the product. These administrative functions typically fall into a number of different categories:

a) Management functions that relate to control over the specific TSP enforced by the TOE. For example, definition and update of per-user policy attributes (such as user clearance), known system access control labels, control and management of user groups.

b) Management functions that relate to accountability and authentication controls enforced by the TOE. For example, definition and update of user security characteristics (e.g., unique identifiers associated with user names, user accounts, system entry parameters) or auditing system controls (e.g., selection of audit events, management of audit trails, audit trail analysis, and audit report generation).

c) Management functions that relate to controls over availability. For example, definition and update of availability parameters or resource quotas.

d) Management functions that relate to general installation and configuration. For example, TOE configuration, manual recovery, installation of TOE security fixes (if any), repair and reinstallation of hardware.

e) Management functions that relate to routine control and maintenance of TOE resources. For example, enabling and disabling peripheral devices, mounting of removable storage media, backup and recovery of user and system objects.

915 The first three categories of these functions are specific to the type of policy included in the PP/ST. Accordingly, the families that address these functions are included in the policy-specific groups. The latter two categories, however, are independent of policy and are addressed by this family.

916 Note that all of these functions need to be present in a TOE, depending on the families included in the PP or ST. It is the responsibility of the PP/ST author to ensure that adequate functions will be provided to manage the system in a secure fashion.

FPT_TSM.1 Management Functions

User Application Notes

917 This component addresses the threat that inadequate capabilities have been provided to the administrator for system management, and partially addresses the threat of external disaster by providing backup capabilities.

918 For FPT_TSM.1.1, it is the responsibility of the PP/ST author to identify the significant TSF configuration parameters that must be settable by the administrator.

Determination of any additional parameters that may be settable by the administrator is up to the developer.

919 For FPT_TSM.1.2, it is the responsibility of the PP/ST author to identify the minimal set of peripherals that must be capable of being enabled or disabled by the administrator. Determination of any additional peripherals that are capable of being enabled or disabled by the administrator is up to the developer. It is also the responsibility of the PP/ST author to identify the significant TSF data and objects within the TSC that must be capable of being backed up and restored. Determination of any additional TSF data and objects within the TSC that are capable of being backed-up or restored is up to the developer.

Evaluator application notes

920 The developer is allowed to include additional items in any of the sets subject to assignment.

921 It is acceptable for many of the security administration functions to be available only in a maintenance or off-line mode. Controls should be in place to limit access during these modes to authorised administrators.

Operations

Assignment:

922 **For FPT_TSM.1.1, the PP/ST author must identify any significant *TSF configuration parameters* that shall be settable by an authorised administrator or state "None."**

Assignment:

923 **For FPT_TSM.1.2, the PP/ST author must specify a *list of desired administrative functions* that the TSF shall provide. These functions may include:**

- **installation and initial configuration of the TSF**
- **enabling and disabling peripheral devices**
- **start-up and shutdown**
- **back-up and restore capabilities (and the list of object types that must be covered by back-up capabilities)**

FPT_TST TSF Self Test

924 The family defines the requirements for the definition of self-testing of the TSF with respect to some expected correct operation. Examples are calls to enforcement functions, sample arithmetical operations on a critical parts of the TOE, etc. These tests can be carried out either in some maintenance state, at start-up, on-line, or continuously. The actions to be taken by the TOE as the result of self testing are defined in other families.

User notes

925 The term “correct operation of the TSF” refers primarily to the operation of the TSF software and the integrity of the TSF data. The abstract machine upon which the TSF software is implemented is tested via dependency on FPT_AMT.

Documentation notes

926 The indicated assurance documentation (if applicable) shall contain the indicated information:

a) A description of the product features that can be used to periodically demonstrate the correct operation of the TSF [AGD_ADM Administrator guidance].

FPT_TST.1 Periodic TSF Testing

User Application Notes

927 This component provides support for the periodic testing of the critical functions of the TSF’s operation by requiring the ability for an authorised administrator to periodically invoke testing functions.

Evaluator application notes

928 It is acceptable for the functions that are available to the authorised administrator for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access during these modes to authorised administrators.

FPT_TST.2 TSF Testing During Start-Up

User Application Notes

929 This component adds to the support for the periodic testing of the critical functions of the TSF by calling for not only the ability for an authorised administrator to periodically invoke the tests, but to have tests executed as part of the TSF start-up mechanism.

Evaluator application notes

930 It is acceptable for the functions that are available to the authorised administrator for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access during maintenance to authorised administrators.

FPT_TST.3 TSF Testing During Normal Operation

User Application Notes

931 This component adds to the support for the periodic testing of the TSF by calling for not only the ability for an authorised administrator to periodically invoke the tests and have the tests executed as part of the TSF start-up mechanism, but to have the TSF periodically perform the tests during normal operation.

Evaluator application notes

932 It is acceptable for the functions that are available to the authorised administrator for periodic testing to be available only in an off-line or maintenance mode. Controls should be in place to limit access during maintenance to authorised administrators.

FPT_TSU TOE Administrative Safe Use

933 The components of this family address general characteristics of TSF administrative interfaces that reduce the likelihood that an unskilled authorised administrator will use a TSF interface in an insecure manner. To some extent, these components address ease of use of the administrative function; however, ease of use is a subjective measure that cannot be precisely measured or evaluated.

User notes

934 The components in this family support the administrator in selecting valid parameters. In AVA_MSU, the requirements are provided to ensure that the administrator knows whether the system is secure with the selected parameters.

Evaluator notes

935 Not all statements of legal or sensible values called out in Administrator Guidance can be checked or enforced by software. Some checks require knowledge beyond semantics and border on the realm of fuzzy logic and artificial intelligence. The judgement of whether a check of legal input value is enforceable must be left up to the skill of the evaluators.

Documentation notes

936 The indicated assurance documentation (if applicable) shall contain the indicated information:

a) Any range constraints or specification of valid input values to be accepted by security relevant administrator commands [AGD_ADM Administrator guidance].

FPT_TSU.1 Enforcement of Administrative Guidance

937 This component addresses problems that may result from boundary conditions being violated.

User Application Notes

938 This component should be used in those situations where the PP/ST author wants to ensure that administrative restrictions are checked whenever possible. It addresses the threat of the unskilled authorised administrator providing out-of-range values through the administrative interface.

FPT_TSU.2 Safe Administrative Defaults

User Application Notes

939 This component should be used whenever there is the risk of unskilled administrators. It addresses the threat of the unskilled authorised administrator by

not only enforcing bounds restrictions, but by reducing the need to guess parameters for selected commands (as defaults are provided).

940 It is the responsibility of the developer to determine the value for each default parameter in the absence of guidance elsewhere in the PP/ST.

Evaluator application notes

941 A default for a parameter is a value that is used for that parameter when null input is provided for that parameter through the TSF interface. For example: typing <return> to a prompt; giving 0 or a null pointer through a function call.

942 A safe default is a default value that, if used, will not compromise the TSP of the TSF. The safe default should be the most restrictive default with respect to the TSP.

Operations

Assignment:

943 **For FPT_TSU.2, the PP/ST author shall specify a *list of security attributes* for which the TSF must provide safe defaults.**

FPT_TSU.3 Administrator Defined Defaults

User Application Notes

944 This component provides the authorised administrator with the ability to specify the default values to be used by the TSF.

Evaluator application notes

945 A default for a parameter is a value that is used for that parameter when null input is provided for that parameter through the TSF interface. For example: typing <return> to a prompt; giving 0 or a null pointer through a function call.

946 A safe default is a default value that, if used, will not compromise the TSP of the TSF. The safe default should be the most restrictive default with respect to the TSP.

Operations

Assignment:

947 **For FPT_TSU.3, the PP/ST author shall specify a *list of security attributes* for which the TSF must provide safe defaults.**

Assignment:

948 **For FPT_TSU.3, the PP/ST author should specify a *list of security attributes* for which the authorised administrator is allowed to modify the TSF provided defaults.**

Class FRU

Resource Utilisation

949 This class provides three families which support availability of required resources, such as processing capability when needed. The family Fault Tolerance provides protection against unavailability of capabilities caused by failure of the TOE. The family Priority of Service ensures that the resources will be allocated to the more important or time-critical tasks and cannot be monopolised by lower priority tasks. The family Resource Allocation ensures that users have limits on their minimum and maximum available resources, and therefore cannot crowd out other users.

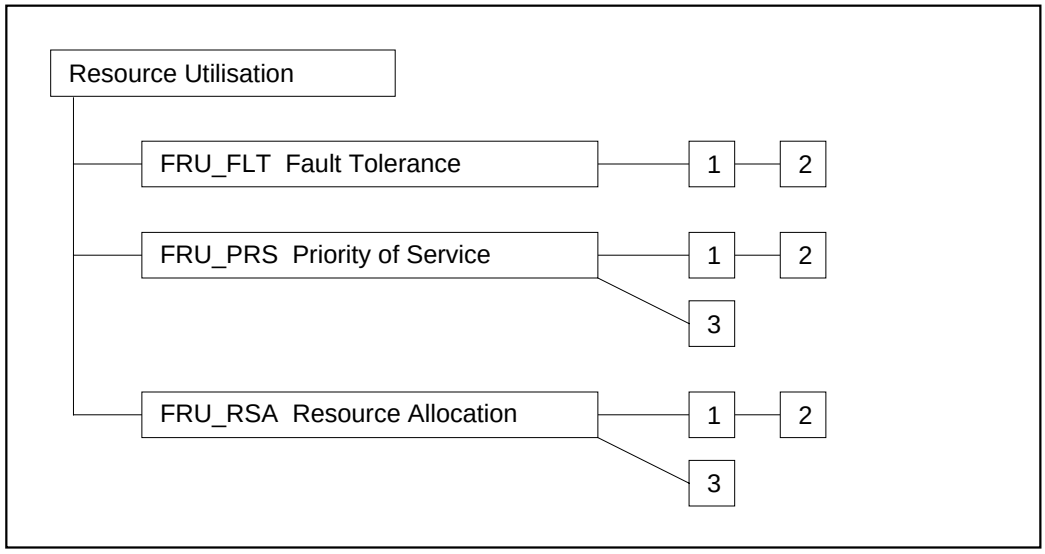

Figure 1.15 - Resource Utilisation class decomposition

FRU_FLT Fault Tolerance

950 This family provides requirements for the availability of capabilities even in the case of failures. Examples of such failures are power failure, hardware failure, or software error. In case of these errors, if so specified, the TOE will maintain the specified capabilities. The PP/ST author could specify, for example, that a TOE used in a nuclear plant will continue the operation of the shut-down procedure in the case of power-failure, or communication-failure.

User notes

951 Since the TOE can only continue its correct operation if the TSP is enforced, there is a requirement that the system must remain in a secure state after a failure. This capability is provided by FPT_FLS.1.

952 The mechanisms to provide fault tolerance could be active or passive. In case of an active mechanism, specific functions are in place which are activated in case the error occurs. For example, a fire alarm is an active mechanism; the TSF will detect the fire and can take action such as switching operation to a backup. In a passive scheme, the architecture of the TOE is capable of handling the error. For example, the use of a majority voting scheme with multiple processors is a passive solution; failure of one processor will not disrupt the operation of the TOE (although it need to be detected to allow correction).

953 For this family, it does not matter whether the failure has been initiated accidentally (such as flooding or unplugging the wrong device) or intentionally (such as monopolising).

FRU_FLT.1 Degraded Fault Tolerance

User Application Notes

954 This component is intended to specify which capabilities the system will still provide after a failure of the system. Since it would be difficult to describe all specific failures, categories of failures may be specified. Examples of general failures are flooding of the computer room, short term power interruption, breakdown of a CPU or host, software failure, overflow of buffer.

Operations

Assignment:

955 **In FRU_FLT.1.1 the PP/ST author should specify which [*list of capabilities*] the TOE will maintain during and after a specified failure.**

Assignment:

956 **In FRU_FLT.1.1 the PP/ST author should specify the [*list of type of failures*] which the TOE explicitly has to be protected against. If a failure in this list occurs the TOE will be able to continue its operation.**

FRU_FLT.2 Limited Fault Tolerance

User Application Notes

957 This component is intended to specify against what type of failures the system must be resistant. Since it would be difficult to describe all specific failures, categories of failures may be specified. Examples of general failures are flooding of the computer room, short term power interruption, breakdown of a CPU or host, software failure, or overflow of buffer.

Operations

Assignment:

958 **In FRU_FLT.2.1 the PP/ST author should specify the [*list of types of failures*] which the TOE explicitly has to be protected against. If a failure in this list occurs the TOE will be able to continue its operation.**

FRU_PRS Priority of Service

959 The requirements of this family allow the TSF to control the use of resources within the TSC by users and subjects such that high priority activities within the TSC will always be accomplished without interference or delay due to low priority activities. In other words, time critical tasks will not be delayed by tasks which are less time critical.

960 This family could be applicable to several types of resources, for example, processing capacity, and communication channel capacity.

961 The Priority of Service mechanism might be passive or active. In a passive Priority of Service system, the system will select the task with the highest priority when given a choice between two waiting applications. While using passive Priority of Service mechanisms, when a low priority task is running, it cannot be interrupted by a high priority task.While using an active Priority of Service mechanisms, lower priority tasks might be interrupted by new high priority tasks.

User notes

962 There are different components for the Priority of Service family and for the management of these components. So, for example, a PP/ST author can select not to use the management functionality, but use default values for the priority of types of subjects. For example ‘0’ for all time-critical subjects, ‘1’ for subjects that interact with the user but are not time-critical, and ‘2’ for all others. In such cases the PP/ST author might wish to consider including the FPT_TSM.1 component.

963 The audit requirement states that all reasons for rejection should be audited. It is left to the developer to argue that an operation is not rejected but delayed.

Documentation notes

964 The indicated administrative guidance (if applicable) shall contain the indicated information:

a) Information on how users are to use the Priority of Service functions [AGD_USR User guidance]

b) Information on how administrators are to set up and configure the Priority of Service [AGD_ADM Administrator guidance]

FRU_PRS.1 Limited Priority of Service

User Application Notes

965 This component defines priorities for a subject, and the resources for which this priority will be used. If a subject attempts to take action on a resource controlled by the Priority of Service requirements, the access and/or time of access will be dependent on the subject’s priority, the priority of the currently acting subject, and the priority of the subjects still in the queue.

Operations

Assignment:

966 **For FRU_PRS.1.2, the PP/ST author should specify the list of [*controlled resources*] for which the TSF enforces priority of service (e.g. resources such as processes, disk space, memory, bandwidth).**

FRU_PRS.2 Full Priority of Service

User Application Notes

967 This component defines priorities for a subject. All shareable resources in the TSC will be subjected to the Priority of Service mechanism. If a subject attempts to take action on a shareable TSC resource, the access and/or time of access will be dependent on the subject's priority, the priority of the currently acting subject, and the priority of the subjects still in the queue.

FRU_PRS.3 Priority of Service Management

User Application Notes

968 This component provides management of the Priority of Service capability by an authorised administrator.

FRU_RSA Resource Allocation

969 The requirements of this family allow the TSF to control the use of resources within the TSC by users and subjects such that unauthorised denial of service will not take place by means of monopolisation of resources by other users.

User notes

970 Resource allocation rules allow the creation of quotas or other means of defining limits on the amount of resource space or time that may be allocated on behalf of a specific user. These rules may be, for example:

- Provide for object quotas that constrain the number and/or size of objects a specific user may allocate.
- Control the allocation/deallocation of preassigned resource units where these units are under the control of the TSF.

971 In general, these functions will be implemented through the use of attributes assigned to users and resources.

Documentation notes

972 The indicated administrative guidance (if applicable) shall contain the indicated information:

a) Information on how users are to use the Resource allocation [AGD_USR User guidance].

b) Information on how administrators are to set up and configure the resource allocation [AGD_ADM Administrator guidance]

FRU_RSA.1 Maximum Quotas

User Application Notes

973 This component provides requirements for quota mechanisms that apply to only some of the shareable resources in the TOE. The requirements allow the quotas to be associated with a user, or possibly assigned to groups of users as applicable to the TOE.

Operations

Assignment:

974 **In FRU_RSA.1.1, the PP/ST author should specify the list of *controlled resources* for which resource allocation limits are required (e.g. processes, disk space, memory, bandwidth). If all resources in the TSC need to be included the words “all TSC resources” can be specified.**

Selection:

975 **In FRU_RSA.1.1, the PP/ST author should select whether the maximum quotas apply to *individual users or to a defined group of users* or both.**

Selection:

976 **In FRU_RSA.1.1, the PP/ST author should select whether the maximum quotas can be used at the *simultaneously or whether they apply to a period of time* in which they can be used.**

FRU_RSA.2 Minimum and Maximum Quotas

User Application Notes

977 This component provides requirements for quota mechanisms that apply to all of the shareable resources in the TOE. The requirements allow the quotas to be associated with a user, or possibly assigned to groups of users as applicable to the TOE.

Operations

Assignment:

978 In FRU_RSA.2.1, the PP/ST author should specify the *controlled resources* for which maximum resource allocation limits are required (e.g. processes, disk space, memory, bandwidth). If all resources in the TSC need to be included the words “all TSC resources” can be specified.

Selection:

979 In FRU_RSA.2.1, the PP/ST author should select whether the maximum quotas apply to *individual users or to a defined group of users* or both.

Selection:

980 In FRU_RSA.2.1, the PP/ST author should select whether the maximum quotas can be used at the *simultaneously or whether they apply to a period of time* in which they can be used.

Assignment:

981 **In FRU_RSA.2.2, the PP/ST author should specify the *controlled resources* for which a minimum allocation limit needs to be set (e.g. processes, disk space, memory, bandwidth). If all resources in the TSC need to be included the words “all TSC resources” can be specified.**

Selection:

982 **In FRU_RSA.2.2, the PP/ST author should select whether the minimum quotas apply to *individual users or to a defined group of users* or both.**

Selection:

983 **In FRU_RSA.2.2, the PP/ST author should select whether the minimum quotas can be used at the *simultaneously or whether they apply to a period of time* in which they can be used.**

FRU_RSA.3 Quota Management

User Application Notes

984 This component provides requirements for managing the quota mechanism on the individual user level.

Operations

Selection:

985 **In FRU_RSA.3.1, the PP/ST author should select whether the management of the quotas apply to *individual users or to a defined group of users* or both.**

Class FTA

TOE Access

986 The establishment of a user's session typically consists of the creation of one or more subjects that perform operations in the TOE on behalf of the user. At the end of the session establishment procedure, provided the TOE access requirements are satisfied, the created subjects bear the attributes determined by the identification and authentication functions.

987 A user session is defined as the period starting at the time of the authentication up to the moment that all subjects (resources and attributes) have been deallocated.

988 Figure 1.16 shows the decomposition of this class into its constituent components.

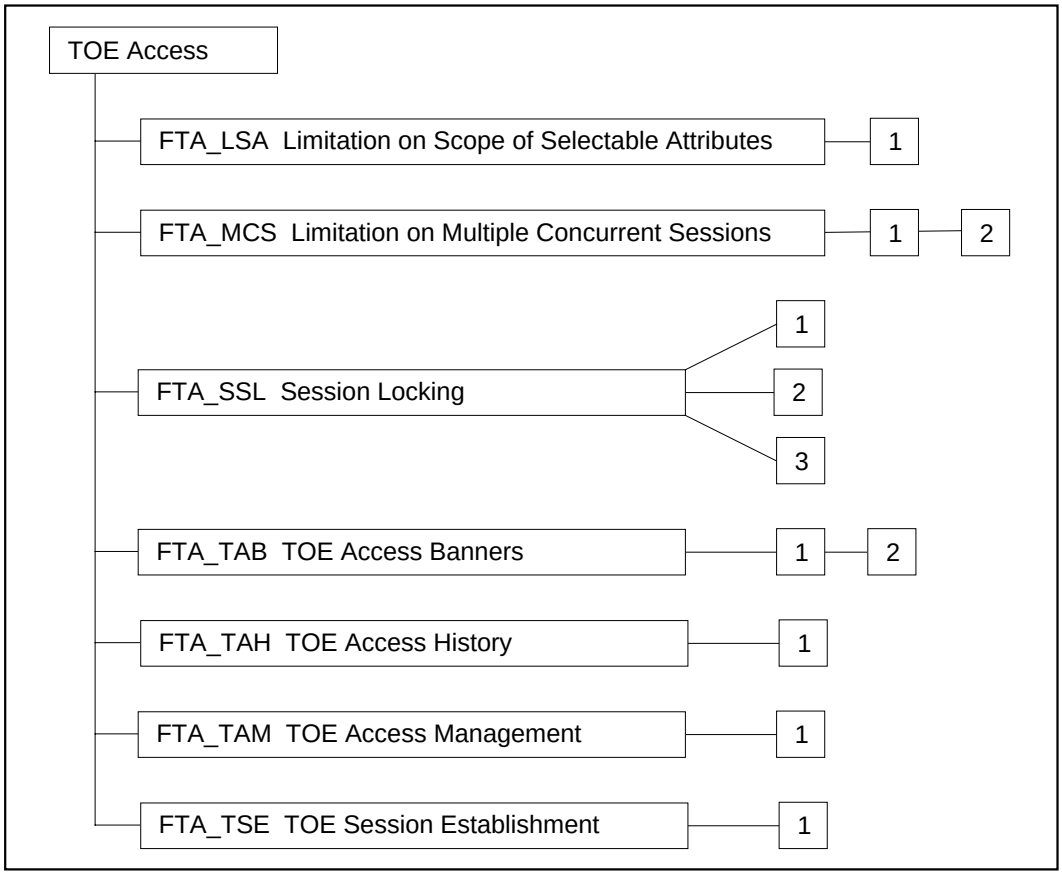

Figure 1.16 - TOE Access class decomposition

FTA_LSA Limitation on Scope of Selectable Attributes

989 This family defines requirements that will limit the attributes a user may select, and the subjects to which a user may be bound, based on: the method of access; the location or port of access; and/or the time (e.g., time-of-day, day-of-week).

User notes

990 This family provides the capability for PP/ST authors to specify requirements for the TSF to place limits of the domain of an authorised user's security attributes based on an environmental condition. For example, a user may be allowed to establish a secret session during normal business hours but outside those hours the same user may be constrained to only establishing unclassified sessions. The identification of relevant constraints on the domain of selectable attributes can be achieved through the use of the selection operation. These constraints can be applied on an attribute-by-attribute basis. When there exists a need to specify constraints on multiple attributes this component will have to be replicated multiple times for each attribute. Attributes limitations can be specified in terms of any combination of the following parameters:

a) The method of access can be used to specify in which type of environment the user will be operating (e.g., file transfer protocol, terminal, vtam).

b) The location of access can be used to constrain the domain of a user's selectable attributes based on a user's location or port of access. This capability is of particular use in environments where dial-up facilities or network facilities are available.

c) The time of access can be used to constrain the domain of a user's selectable attributes. For example, ranges may be based upon time-of-day, day-of-week, or calendar dates. This constraint provides some operational protection against user actions that could occur, at a time where proper monitoring or where proper procedural measures may not be in place.

Documentation notes

The indicated assurance documentation (if applicable) should contain the following information:

a) Specification of the manner in which the TSF places limits on the security attributes that a user may select and the subjects a user may be bound to, based upon how the session is initiated [ADV_FSP Functional specification].

b) Guidance on the meaning of this TOE access parameter, and guidance regarding intelligent selection of values [AGD_ADM Administrator guidance].

c) Guidance on how to specify limitations on the attributes that a user may select based on the method of access [AGD_ADM Administrator guidance].

FTA_LSA.1Limitation on Scope of Selectable Attributes

Operations

Assignment:

991 **In FTA_LSA.1.1 the PP/ST author should specify the set of *session security attributes* which could be constrained. Examples of these session security attributes are user clearance level, and user integrity level, role.**

Assignment:

992 **In FTA_LSA.1.1 the PP/ST author should specify the set of *attributes* the authorised administrator can use to determine the scope of the session security attributes. Examples of such attributes are user identity, originating location, time of access, and method of access.**

FTA_MCS Limitation on Multiple Concurrent Sessions

993 This family defines how many sessions a user can have at the same time (concurrent sessions). This number of concurrent sessions can either be set for a group of users or for each individual user.

Documentation notes

The indicated assurance documentation (if applicable) should contain the following information:

a) Specification of the manner in which the TSF places limits on the number of concurrent sessions (per user attribute for FTA_MCS.2) [ADV_FSP Functional specification].

b) Guidance on the meaning of this TOE access parameter, and guidance regarding selection of values [AGD_ADM Administrator guidance].

c) Guidance on how to specify the number of allowed concurrent sessions [AGD_ADM Administrator guidance].

FTA_MCS.1 Basic Limitation on Multiple Concurrent Sessions

User Application Notes

994 This component allows an authorised administrator to place limits on the number of concurrent sessions that users are able to invoke. This provides protection against actions that cannot be properly monitored or where procedural measures cannot be properly put in place.

FTA_MCS.2 Per User Attribute Limitation on Multiple Concurrent Sessions

User Application Notes

995 This component provides an authorised administrator with additional capabilities over those of FTA_MCS.1, by allowing further constraints to be placed on the number of concurrent sessions that users are able to invoke. These constraints are in terms of a user's security attributes, such as a user's identity, or membership to a role. This provides protection against actions that cannot be properly monitored or where procedural measures cannot be properly put in place.

Operations

Assignment:

996 **For FTA_MCS.2.1 the PP/ST author should specify the *security attributes* the authorised administrator can use to specify the maximum number of sessions per user.**

Selection:

997 **For FTA_MCS.2.3 the PP/ST author should specify whether the *minimum number of sessions* shall be selected or the *maximum number of sessions,* when there is a conflict between the rules specified by the authorised administrator. The PP/ST author should specify one and only one of the options.**

FTA_SSL Session Locking

998 This family defines requirements for the TSF to provide the capability for locking and unlocking of interactive sessions (e.g., keyboard locking).

999 When a user is directly interacting with subjects in the TOE (interactive session), the user's terminal is vulnerable if left unattended. This family provides requirements for the TSF to disable (lock) the terminal or terminate the session after a specified period of inactivity, and for the user to initiate the disabling (locking) of the terminal. To reactivate the terminal, the user must authenticate himself to the TSF.

1000 A user is considered inactive, if he has not provided any stimulus to the TOE for a period of time, specified by an authorised administrator.

1001 A PP/ST author should consider whether FTP_TRP.1 Trusted Path should be included. In that case, the function 'session locking' should be included in the operation in FTP_TRP.1.

Documentation notes

The indicated assurance documentation (if applicable) should contain the following information:

a) Specification of the manner in which the TSF locks or terminates a user session [ADV_FSP Functional specification].

b) Guidance on how to specify the interval of user inactivity before locking or terminating the session, and guidance on how to unlock a session [AGD_ADM Administrator guidance].

c) Guidance to users on how to lock and unlock a session [AGD_USR User guidance].

FTA_SSL.1 TSF-Initiated Session Locking

User Application Notes

1002 FTA_SSL.1 TSF-Initiated Session Locking, provides the capability for the TSF to lock an active user session after an authorised-administrator-specified period of time. Locking and terminating a session are viewed as being functionally equivalent for Protection Profiles. Locking a terminal would prevent any further interaction with an existing active session through the use of the locked terminal, while terminating the session altogether would also prevent further use of the active session.

FTA_SSL.2 User-initiated Locking

User Application Notes

1003 FTA_SSL.2 User-initiated Locking, provides the capability for an authorised user to lock and unlock his/her own terminal. This would provide authorised users with the ability to effectively block further use of their active sessions without having to terminate the active session.

FTA_SSL.3 TSF-initiated Termination

User Application Notes

1004 FTA_SSL.3 TSF-initiated Termination, requires that the TSF shall terminate an interactive user session after a period of inactivity.

1005 The PP/ST author should be aware that a session may continue after the user terminated its activity, for example background processing. This requirement would terminate this background subject after a period of inactivity of the user without regard to the status of the subject.

FTA_TAB TOE Access Banners

1006 Prior to identification and authentication, TOE access requirements provide the ability for the TOE to display an advisory warning message to potential users pertaining to appropriate use of the TOE.

User notes

1007 FTA_TAB.1 provides banner protection by default, that cannot be modified. FTA_TAB.2 provides the additional capability to specify site or organisation specific banners.

Documentation notes

The indicated assurance documentation (if applicable) should contain the following information:

a) Guidance on how to configure the TOE access banner message per the FTA_TAB.2 Configurable TOE Access Banners component. [AGD_ADM Administrator guidance]

FTA_TAB.1 Default TOE Access Banners

Operations

Assignment:

1008 **In FTA_TAB.1.2 the PP/ST author should specify the [*warning message*], which will be displayed prior to TOE access. For example, the following message could be assigned:**

"NOTICE: This is a private computer system. All users of this system are subject to having their activities audited. Anyone using this system consents to such auditing. All unauthorised entries or actions revealed by this auditing can be used as evidence and may lead to criminal prosecutions."

FTA_TAB.2 Configurable TOE Access Banners

User Application Notes

1009 This component allows the authorised administrator to change the default warning.

Operations

Assignment:

1010 **In FTA_TAB.2.2 the PP/ST author should specify the default [*warning message*], which will be displayed prior to TOE access. For example, the following message could be assigned:**

1011 **"NOTICE: This is a private computer system. All users of this system are subject to having their activities audited. Anyone using this system consents to such auditing. All unauthorised entries or actions revealed by this auditing can be used as evidence and may lead to criminal prosecutions."**

FTA_TAH TOE Access History

1012 This family defines requirements for the TSF to display to users, upon successful session establishment to the TOE, a history of successful and unsuccessful attempts to access the account. This history may include the date, time, means of access, and port of the last successful access to the TOE, as well as the number of successful, and unsuccessful attempts to access the TOE since the last successful access by the identified user.

User notes

1013 This family can provide authorised users with information that may indicate the possible misuse of their user account.

Documentation notes

The indicated assurance documentation (if applicable) should contain the following information:

a) Specification of the manner in which the TSF displays the TOE access history message to the user [ADV_FSP Functional specification].

b) Guidance to users on how to use and understand the information presented to them regarding TOE access history [AGD_USR User guidance].

FTA_TAH.1 TOE Access History

Operations

Selection:

1014 **In FTA_TAH.1.1, the PP/ST author should select the security attributes of the last successful session establishment that will be shown on the screen. The items are: date, time, method of access (such as ftp), and/or location (e.g., terminal 50).**

Selection:

1015 **In FTA_TAH.1.2, the PP/ST author should select the security attributes of the last unsuccessful session establishment that will be shown on the screen. The items are: (date, time, method of access (such as ftp), and/or location (e.g., terminal 50).**

FTA_TAM TOE Access Management

1016 This family defines requirements to specify, remove or inspect TOE access parameters by an authorised administrator. This family defines requirements for the setup of any TOE access security policy.

Documentation notes

1017 The indicated assurance documentation (if applicable) should contain the indicated information:

a) Guidance on how to use the TOE access management function [AGD_USR User guidance and AGD_ADM Administrator guidance].

b) Specification of the TSF provided TOE access management function [ADV_FSP Functional specification].

FTA_TAM.1 Basic TOE Access Management

User Application Notes

1018 For this component, authorised administrators are permitted to display or modify TOE access parameters.

Operations

Selection:

1019 **In FTA_TAM.1.2 the PP/ST author should be able to specify whether the authorised administrators shall be able to review the access parameters for a user or for all users with a specified access parameter setting, or both.**

FTA_TSE TOE Session Establishment

1020 This family provides the ability to place constraints on the establishment of a user session. These constraints can be specified in terms of a user's attributes such as the user identity, role, or confidentiality level.

1021 This family defines requirements to allow or deny an authorised user to establish a session with the TOE based on attributes such as the location or port of access, the user's security attribute (e.g., identity, clearance level, integrity level membership in a role), ranges of time (e.g., time-of-day, day-of-week, calendar dates) or combinations of parameters.

User notes

1022 This family provides the capability for PP/ST author to specify requirements for the TOE to place constraints on the ability of authorised user to establish a session with the TOE. The identification of relevant constraints can be achieved through the use of the selection operation. Session establishment constraints can be specified in terms of any combination of the following parameters:

a) The location of access can be used to constrain the ability of a user to establish an active session with the TOE, based on the user's location or port of access. This capability is of particular use in environments where dial-up facilities or network facilities are available.

b) The mode of access can be used to place constraints on the ability of a user to establish an active session with the TOE, based on the user's authorised security attributes. For example, these attributes would provide the capability for an administrator to allow or deny session establishment based on any of the following:

- a user's identity;
- a user's clearance level;
- a user's integrity level; and
- a user's membership in a role.

This capability is particularly relevant in situations where authorisation may take place at a different location where TOE access checks are performed.

c) The time of access can be used to constrain the ability of a user to establish an active session with the TOE based on ranges of time. For example, ranges may be based upon time-of-day, day-of-week, or calendar dates. This constraint provides some operational protection against user actions that could occur at a time where proper monitoring or where proper procedural measures may not be in place.

d) In addition, this family provides the capability for an authorised administrator to specify TOE session establishment constraints that are based on a combination of a user's selected attributes, location of access, time of access and the location or port of access.

Documentation notes

The indicated assurance documentation (if applicable) should contain the following information:

a) Specification of the manner in which the TSF places limits on the establishment of a user session. [ADV_FSP Functional specification].

b) Guidance on the meaning of relevant parameters, and guidance regarding intelligent selection of values [AGD_ADM Administrator guidance].

c) Guidance on how to specify session establishment constraints [AGD_ADM Administrator guidance].

FTA_TSE.1 TOE Session Establishment

Operations

Assignment:

1023 **In FTA_TSE.1.1 the PP/ST author should specify the [*attributes*] that can be used to restrict the session establishment. Example of possible attributes are user identity, originating location (e.g. no remote terminals), time of access (e.g. outside hours), or method of access (e.g. X-windows).**

Class FTP

Trusted Path/Channels

1024 Users often need to perform functions through direct interaction with the TSF. A trusted path ensures that a user is communicating directly with the TSF whenever it is invoked. Trusted path is usually desired for initial login, but may also be desired at other times during a user's session. Trusted path exchanges may be initiated by a user or the TSF. A user's response via the trusted path guarantees that untrusted applications cannot intercept or modify the user's response.

1024 Figure 1.17 illustrates the relationships between the various types of communication that may occur within a TOE or network of TOEs (i.e., Internal TOE transfers, Inter-TSF transfers, and Import/Export Outside of TSF Control) and the various forms of Trusted Path.

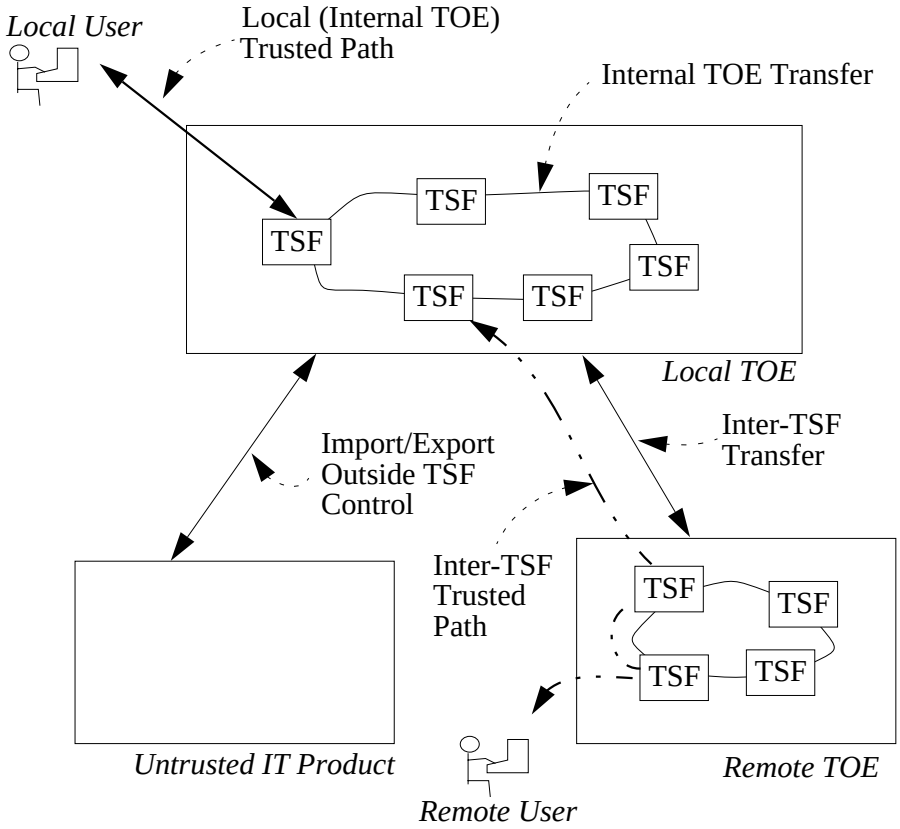

Figure 1.17 - Trusted Paths and Trusted Channels

1025 Absence of a trusted path may allow breaches of accountability or access control in environments where untrusted applications are used. These applications can intercept user-private information, such as passwords, and use it to impersonate other legitimate users. As a consequence, responsibility for any system actions cannot be reliably assigned to an accountable entity. Also, these applications could output erroneous information on an unsuspecting user's display, resulting in subsequent user actions that may be erroneous and may lead to a security breach.

1026 Figure 1.18 shows the decomposition of this class into its constituent components.

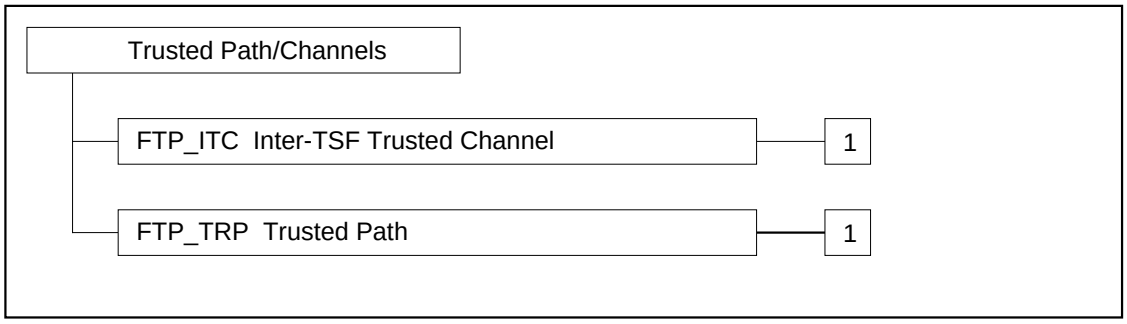

Figure 1.18 - Trusted Path / Channels Class decomposition

FTP_ITC Inter-TSF Trusted Channel

1027 This family defines the rules for the creation of a trusted channel connection that goes between the TSF and other TSFs for the performance of security critical operations between TSFs. Examples of such security critical operations may include the updating of an authentication database on one TSF by another or the transfer of audit data to a TSF whose function is the collection of audit data. This family should be included whenever there are requirements for the secure communication of user or TSF data between two TSFs.

Documentation notes

1028 The indicated assurance documentation (if applicable) should contain the indicated information:

a) Information on how administrators are to set up and configure the trusted channel [AGD_ADM Administrator guidance].

FTP_ITC.1 Inter-TSF Trusted Channel

User Application Notes

1029 This component should be used when a trusted communication channel between two TSFs is required.

Operations

Selection:

1030 **In FTP_ITC.1.2, the PP/ST author must specify whether the *local TSF, the remote TSF, or both* shall have the capability to initiate the trusted channel.**

Assignment:

1031 **In FTP_ITC.1.3, the PP/ST author should specify the *functions for which a trusted channel is required*. Examples of these functions may include: transfer of user, subject, and/or object security attributes and ensuring consistency of TSF data.**

FTP_TRP Trusted Path

1032 This component defines the requirements to establish and maintain trusted communication to or from human users and the TSF. A trusted path may be required for any security-relevant interaction. Trusted path exchanges may be initiated by a human user during an interaction with the TSF, or the TSF may establish communication with the human user via a trusted path.

Documentation notes

1033 The indicated assurance documentation (if applicable) should contain the indicated information:

a) Information on how users are to use the trusted path [AGD_USR User guidance].

FTP_TRP.1 Trusted Path

User Application Notes

1034 This component should be used when a trusted communication channel between a local human user and the TSF is required for initial authentication purposes and any additional services, and the channel is initiated by the local human user.

Operations

Selection:

1035 **In FTP_TRP.1.1, the PP/ST author should specify whether the trusted path must be extended to *remote and/or local* users.**

Selection:

1036 **In FTP_TRP.1.2, the PP/ST author should specify whether *the TSF, local users, and/or remote users* should be able to initiate the trusted path.**

Selection:

1037 **In FTP_TRP.1.3, the PP/ST author should specify whether the trusted path is to be used for *initial user authentication and/or for other specified services*.**

Assignment:

1038 **In FTP_TRP.1.3, the PP/ST author should identify *other services for which trusted path is required,* if any.**

Annex B

Guidance for selecting functional security requirements

B.1 Introduction

1039 This document aids in selecting appropriate security functional requirements for Protection Profiles and/or Security Targets by identifying the security functional requirement families relevant for specific protection objectives and threats.

1040 The guidance in this document will aid in achieving an acceptable level of protection for a user's identified threats. This guidance is made in part due to the fact that for many applications there is no comprehensive policy in effect which covers the selection of functional requirements. However, where policy does exist to identify security requirements, this document should not be taken to supersede or override that organisational security policy.

1041 This document contains the following sections:

1042 Section 1.1.2 Family selection describes an overview of the approach for selecting appropriate families of functional components.

1043 Section 1.1.3 Component selection describes the approach for selecting an appropriate functional component within a functional family.

1044 Section B.1.3 Security objectives then provides a synopsis of common security objectives for information technology.

1045 Section 1.3 General threats provides a list of general threats for common IT applications.

1046 Section 1.4 Detailed threats provides a list of detailed threats for which the security functional requirements contained in Part 2 are known to be effective countermeasures.

1047 The user of this document is advised that the information contained at the level of the detailed threats is considered to be more complete and has been more widely reviewed then the threats and relationships found at the level of the general threats.

B.1.1 Family selection

1048 A vulnerability is a design, implementation, or operations flaw that may exist in a system. A threat agent is capable of exploiting a vulnerability, and causing damage to an organisation's protected data or IT resources.

1049 There are many types of threats and vulnerabilities that are known to confront computer systems. Assurance techniques specified in part 3 of the CC provide confidence in a correct implementation of security functional requirements. The security functional requirements must be appropriately selected in order to address the threats that face an organisation. Even the most highly assured systems can be vulnerable to attack, if the implemented security functions do not address the appropriate threats.

1050 The selection of security functional requirement families and the specification of specific security functional requirements are often motivated by the desire to mitigate threats.

1051 Within this guide the selection of applicable functional requirement families may begin with the list of commonly acknowledged security objectives, the list of general threats, or the list of detail threats. A security objective is a statement of intent to counter a specific collection of threats, of which only a subset of those threats may pertain to the TOE. The relationship between security objectives and threats is depicted in figure B.1. The use of double arrows indicates a many-to-many relationship among security objectives and threats. As such, each security objective maps to multiple general threats and each general threat maps back to one or more security objectives.

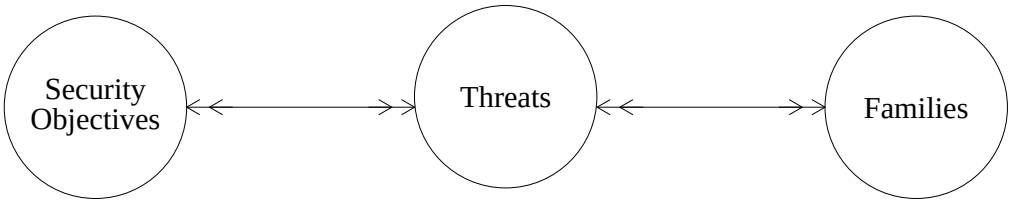

Figure B.1 - Security objectives, threats, and families relationship

1052 The threats in this document are specified at two levels of detail: general threats, and detailed threats; where the detailed threats are specializations of the general threats. For example, a general threat may pertain to “the unauthorised observation of user data” while an associated detailed threat may pertain to a specific instance of the general threat, such as “the unauthorised observation of user data due to the inability of the TSF to remove residual data from public objects after prior use.”

1053 In a small number of cases security objectives will provide sufficient detail for the identification of detailed threats, and as such for the selection of a security functional requirement family.

1054 For general threats that do not directly references functional families, the general threat references a set of detailed threats.

1055 In all cases, detailed threats can be used to identify relevant security functional families that should be selected to provide functions to counter those threats.

B.1.2 Component Selection

1056 Each security functional family contains one or more security functional components, leveled based on scope, granularity, and/or coverage. This levelling can be used as the basis for the selection of an appropriate functional component within a family. Components represent the smallest selectable unit of security functional specification within the CC. The purpose of the component levelling is to provide information to PP or ST authors to aid in selecting an appropriate component once the family has been identified as being a necessary part of their security requirements. The levelling of the functional family describes the components available within the family, and the relationships between these components.

B.1.3 Security objectives

1057 The table provides a list of commonly known security objectives for information technology and an associated set of references to general threats and detailed threats.

S.O #	Security Objective	Related General Threats
S.O.1.	The security functions of the TOE need to be protected from unauthorised modification.	G.T.16, G.T.18, G.T.19
S.O.2.	The security functions of the TOE need to be enforced for all security relevant activities.	G.T.16, D.T.72
S.O.3.	Users of the TOE need to be accountable for their actions.	G.T.1, G.T.3, G.T.6, G.T.7, G.T.8, G.T.10, G.T.15, G.T.16, G.T.17
S.O.4.	Information must be protected from disclosure to unauthorised individuals.	G.T.1, G.T.3, G.T.4, G.T.5, G.T.10, G.T.11, G.T.9, G.T.12, G.T.16, G.T.17
S.O.5.	Information must be protected from corruption or unauthorised modification.	G.T.1, G.T.3, G.T.4, G.T.5, G.T.9, G.T.13, G.T.16, G.T.17

Table B.1 - Security Objectives

S.O #	Security Objective	Related General Threats
S.O.6.	Information must be available for use when needed.	G.T.1, G.T.4, G.T.5, G.T.14, G.T.16, G.T.17, D.T.20, D.T.84, D.T.86
S.O.7.	The security functions of the TOE need to preserve the privacy of its users.	G.T.1, G.T.3, G.T.4, G.T.5, G.T.16, G.T.17
S.O.8.	Users should only be able to access the TOE in accordance with the TOE access policy.	G.T.1, G.T.16, G.T.17,
S.O.9.	The TOE should allow authorised administrators to administer the TOE in a safe manner	G.T.1, G.T.2

Table B.1 - Security Objectives

1.3 General threats

1058 The table below provides a list of general threats for information technology and an associated set of references to detailed threats.

G.T. #	General Threat	Related Detailed Threats or Related Family
G.T.1.	An authorised administrator may not be able to properly administer the TOE in a manner where the TOE can properly enforce the TSP.	D.T.35, D.T.36, D.T.79, D.T.92, D.T.103, D.T.104
G.T.2.	An authorised administrator may misuse his or her privileges.	D.T.78, D.T.83
G.T.3.	A user may gain the unauthorised ability to assume the identity of another user.	D.T.29, D.T.45, D.T.41, D.T.49, D.T.48
G.T.4.	A user's security attribute may become inconsistent with the user's authorised credentials while being perceived as correct by the TSF.	D.T.35, D.T.54, D.T.56, D.T.60, D.T.63, D.T.72, D.T.42, D.T.43, D.T.44, D.T.45, D.T.48, D.T.83, D.T.109

Table B.2 - General Threats

G.T. #	General Threat	Related Detailed Threats or Related Family
G.T.5.	An object's security attribute may become inconsistent with the security characteristics of the information that it contains while being perceived as correct by the TSF.	D.T.45, D.T.60, D.T.63, D.T.59, D.T.72, D.T.83, D.T.89, D.T.91, D.T.93, D.T.94, D.T.99, D.T.102
G.T.6.	Security relevant actions may be incorrectly traced to a human user.	D.T.27, D.T.72, D.T.109
G.T.7.	A violation of the TSP may occur without an appropriate and timely response.	D.T.13, D.T.14, D.T.15, D.T.16, D.T.19, D.T.26, D.T.33, D.T.60, D.T.72, D.T.10, D.T.107
G.T.8.	A violation of the TSP may not be traceable to the human user responsible for the violation.	D.T.14, D.T.15, D.T.16, D.T.18, D.T.19, D.T.20, D.T.21, D.T.22, D.T.23, D.T.24, D.T.27, D.T.11, D.T.72, D.T.25, D.T.17, D.T.53, D.T.107, D.T.109, D.T.111
G.T.9.	A user may violate the TSP due to the lack of information pertaining to the TOE's authorised use.	D.T.1**,** D.T.2
G.T.10.	A user may gain TOE access where proper monitoring, administrative oversight, and/or procedures are not in place.	D.T.6, D.T.56, D.T.72, D.T.32, D.T.37, D.T.39, D.T.50, D.T.52, D.T.73
G.T.11.	Information may be exposed to an environment where proper physical and/or procedural controls are not locally in place.	D.T.3, D.T.5, D.T.8, D.T.9, D.T.66, D.T.72, D.T.32, D.T.37, D.T.39, D.T.50, D.T.52, D.T.89 D.T.90
G.T.12.	A user may gain the ability to observe user data in a manner that is not consistent with the TSP**.**	D.T.28, D.T.5, D.T.11, D.T.56, D.T.8, D.T.38, D.T.40, D.T.46, D.T.47, D.T.51, D.T.58, D.T.72, D.T.89, D.T.90, D.T.94, D.T.95, D.T.97, D.T.100, D.T.105, D.T.109, D.T.112
G.T.13.	A user may gain the ability to alter user data in a manner that is not consistent with the TSP.	D.T.11, D.T.56, D.T.58, D.T.69, D.T.72, D.T.8, D.T.38, D.T.40, D.T.46, D.T.47, D.T.51, D.T.89, D.T.90, D.T.94, D.T.96, D.T.98, D.T.101, D.T.106, D.T.109

Table B.2 - General Threats

G.T. #	General Threat	Related Detailed Threats or Related Family
G.T.14.	A user may gain the ability to consume resources in a manner that is not consistent with the TSP.	D.T.4, D.T.11, D.T.56, D.T.57, D.T.72, D.T.7, D.T.38, D.T.40, D.T.46, D.T.47, D.T.51, D.T.58, D.T.85, D.T.86, D.T.89, D.T.94
G.T.15.	A user may deny the performance of an action.	D.T.87, D.T.88
G.T.16.	A user may alter TSF data in a manner that is not consistent with the TSP.	D.T.17 D.T.33, D.T.56, D.T.57, D.T.67, D.T.72, D.T.8, D.T.30, D.T.31, D.T.74, D.T.75, D.T.102, D.T.94, D.T.109
G.T.17.	A user may observe TSF data in a manner that is not consistent with the TSP.	D.T.17, D.T.28, D.T.34, D.T.56, D.T.65, D.T.66, D.T.68, D.T.30, D.T.31, D.T.70, D.T.72, D.T.8, D.T.41, D.T.73, D.T.75, D.T.109, D.T.110, D.T.112, D.T.113
G.T.18.	A TSF mechanism may be altered or enter an insecure state due to conditions that occur external to the TSF.	D.T.66, D.T.70, D.T.71, D.T.76, D.T.80, D.T.82, D.T.84
G.T.19.	A TSF mechanism may be altered or enter an insecure state due to conditions that occur internal to the TSF.	D.T.61, D.T.62, D.T.64, D.T.77, D.T.81, D.T.82
G.T.20.	A user may be denied otherwise legitimate access to user data or resources.	D.T.55, D.T.72, D.T.85, D.T.86, D.T.86, D.T.89, D.T.90

Table B.2 - General Threats

1.4 Detailed threats

1059 The table below provides a list of detailed threats for information technology and an associated set of references to a set of detailed threats and a set of security

functional requirement families.

D.T. #	Detailed Threat	Related Family
D.T.1.	A user may unwittingly gain unauthorised access to the TOE, due to the inability of the TOE to warn the user as to what constitutes its unauthorised use.	FTA_TAB
D.T.2.	A user may misuse the TOE due to the lack of awareness as to penalties that may apply as a result of their unauthorised use of the TOE.	FTA_TAB
D.T.3.	A user may be active on multiple sessions where proper physical or procedural controls cannot adequately be provided to ensure simultaneous protection of each active session.	FTA_MCS
D.T.4.	A user may exceed an acceptable level of resource utilisation due to multiple concurrent active sessions.	FTA_MCS
D.T.5.	Information may be exposed in an environment where proper physical and/or procedural constraints are not in place, due to the lack control over the method of access (e.g., ftp, rpc, dial-up, local terminal), given the user's security attributes.	FTA_LSA
D.T.6.	An authorised user may gain access to information and/or resources during a period of time where proper monitoring and/or procedural controls are not adequately in place.	FTA_TSE
D.T.7.	An authorised user may gain access to resources during a period of time where the consumption of the resources can not be tolerated.	**FTA_TSE**
D.T.8.	A user may gain an inappropriate capability to access TOE data or TOE resources, due to the user's method of access.	**FTA_TSE**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.9.	Information may be exposed in an environment where proper physical and/or procedural constraints are not in place due to the location of access to the TOE.	FTA_TSE
D.T.10.	A user may be unable to detect unauthorized use of his/her account due to the lack of historical data regarding successful access to the user's account.	FTA_TAH
D.T.11.	A user may gain unauthorised access to an active session due to the exposure of the user to an unattended active terminal.	FTA_SSL
D.T.12.	A user gains unauthorised read access to residual data that has been left by a process after its execution.	FDP_RIP
D.T.13.	A violation of the FSP may be detected by the TSF without a timely and appropriate response.	**FAU_ARP**
D.T.14.	Parties responsible for the correct enforcement of a FSP may remain unaware of the occurrence of violations of the FSP.	**FAU_ARP, FAU_PAD, FAU_PIT**
D.T.15.	A security relevant event may occur and not be recorded by the TSF.	**FAU_GEN**
D.T.16.	Parties responsible for the correct enforcement of a FSP may remain unaware of the occurrence of a security relevant event.	**FAU_GEN**
D.T.17.	A user may gain unauthorised access to security audit attributes.	**FPT_TSA**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.18.	The meaning of an audit event may be lost due to the inability of the TSF to effectively transform the event to a format that is recognized by subsequent processing or storage functions.	**FAU_PRP, FAU_POP**
D.T.19.	Audit information may be lost due to audit storage space exhaustion.	**FAU_PRP, FAU_STG, FAU_SEL**
D.T.20.	Security audit analysis may become less effective due to an excessive volume of audit data.	**FAU_SEL**
D.T.21.	The TOE performance and storage costs involved in performing security audit analysis may become too high to justify its results.	**FAU_SEL**
D.T.22.	A damage assessment may not be possible due to the inability to properly store audit data.	**FAU_STG**
D.T.23.	Audit data may be lost due to system failure.	**FAU_STG**
D.T.24.	A authorized administrative user may not be able to effectively derive the meaning of the captured and stored security relevant events.	**FAU_POR**
D.T.25.	A security violation may be recorded but remain undetected, due to the inability to correlate strings of related events.	**FAU_POR**
D.T.26.	Parties responsible for the correct enforcement of a FSP may remain unaware of the significance of a sequence of seemingly benign real time events.	**FAU_PAD, FAU_PIT, FAU_ARP**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.27.	A user may gain unauthorized access to audit trail data that results in hindering a subsequent investigation of violations of the TSP.	**FAU_PRO**
D.T.28.	A user may gain unauthorized access to audit trail data that results in an unauthorized disclosure of audit data.	**FAU_PRO**
D.T.29.	An unauthorised user may penetrate the TOE by correctly guessing authentication data.	**FIA_AFL**
D.T.30.	A user may gain unauthorized access to user security attributes due to the inability to protect such data.	**FIA_ATP (FTP_?)**
D.T.31.	A user may gain unauthorized access to the attributes associated with a session due to the inability to protect such data.	**FIA_ATP (FTP_?)**
D.T.32.	A user may gain unauthorized access to the TOE due to an inappropriate association of authentication attributes with a user.	**FIA_ADA, FIA_ADP**
D.T.33.	A user may gain unauthorized modify access to authentication data due to the inability of the TOE to regulate access to the authentication attributes.	**FIA_ADA, FIA_ADP**
D.T.34.	A user may gain unauthorized read access to authentication data due to the inability of the TOE to protect this data.	**FIA_ADP**
D.T.35.	An authorised administer may not be able to properly maintain user authentication data.	**FIA_ADA**
D.T.36.	An authorised administrator may not be able to properly maintain user security attributes.	**FIA_ATA**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.37.	A user may gain unauthorized access to the TOE due to the uncontrolled modification of user security attributes.	**FIA_ATA**
D.T.38.	A user may gain unauthorized access to TOE data or TOE resources due to the uncontrolled modification of user security attributes.	**FIA_ATA**
D.T.39.	A user may gain unauthorized access to the TOE due to the exploitation of obsolete user security values.	**FIA_ATA**
D.T.40.	A user may gain unauthorized access to TOE data or TOE resources due to the exploitation of user security values.	**FIA_ATA**
D.T.41.	An adversary may exploit user authentication data due to an inconsistent interpretation of the users' authentication attributes while being transmitted between TSFs.	**FIA_ATC**
D.T.42.	A user may violate the TSP due to an inappropriate association of a security attribute to the user.	**FIA_ATD**
D.T.43.	The strength of a security function will be weakened to an unacceptable level due to an inappropriately TSF supplied or generated secret for which the security function depends.	**FIA_SOS**
D.T.44.	A secret will be accidentally or deliberately misused due to an inappropriately supplied or generated secret.	**FIA_SOS**
D.T.45.	A user's identity may be impersonated due to the inability of the TSF to properly validate the user's identity.	**FIA_UAU**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.46.	A user may gain unauthorized access to TOE data or resources due to the inability of the TSF to properly validate a user's identity.	**FIA_UAU**
D.T.47.	A user may gain unauthorised access to TOE data or resources due to the inability of the TSF to validate the user's identity in a timely manner.	**FIA_UAU**
D.T.48.	A user may gain unauthorised use of authentication data by forging or copying authentication data from another user.	**FIA_UAU**
D.T.49.	A user may gain the identity of another user by taking over the user's authenticated session.	**FIA_UAU**
D.T.50.	A user may gain unauthorized access to the TOE by forging or copying authentication data from another user	**FIA_UAU**
D.T.51.	A user may gain unauthorised access to TOE data or resources due to the inability to uniquely identify a user.	**FIA_UID**
D.T.52.	A user may gain unauthorized access to the TOE due to the inability to uniquely identify a user.	**FIA_UID**
D.T.53.	An enterprise may not be able to hold a user accountable for their actions due to the inability to uniquely identify a user.	**FIA_UID**
D.T.54.	A user's security attributes that are attached to a subject may become inappropriately association another subject.	**FIA_USB**
D.T.55.	A user may loss legitimate access capabilities due to the loss of the user's security attributes associated with subject acting on the users' behalf.	**FIA_USB**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.56.	A user may gain inappropriate access capabilities or loss legitimate access capabilities due to an inappropriate modification of a subject's attributes.	**FIA_USB**
D.T.57.	A user may gain unauthorised access to TSF data when it is transferred between parts of the TOE across an internal channel.	**FPT_ITT**
D.T.58.	A user may gain unauthorised access to user data or resources due to the inability of the TOE to effectively revoke attributes.	**FPT_REV**
D.T.59.	The TSF may receive data from an unauthorised user that is perceived by the TSF as being authentic.	**FPT_RPL**
D.T.60.	The validity of a security attribute may change while being perceived by the TSF as being correct.	**FPT_SAE**
D.T.61.	The states of two parts of the TSF may become inconsistent resulting in the failure of the TSF to function as a cohesive whole.	**FPT_SSP**
D.T.62.	The states of two parts of the TSF may become inconsistent due to the inability of the parts to address timing issues.	**FPT_STM**
D.T.63.	Security attributes that are share between TSFs may become inconsistent.	**FPT_TDC**
D.T.64.	An internal failure of the underlying abstract machine may go undetected by the TSF resulting in a TSP violation.	**FPT_AMT**
D.T.65.	The TSF may become inoperative due to the inability of the TSF to replicate TSF data with cooperating TSFs.	**FPT_TRC**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.66.	A violation of the TSF may occur due to the failure of the TOE.	**FPT_FLS**
D.T.67.	An outsider may interfere with TSF data moving between cooperating TSFs in such a way that the data is no longer available as required.	**FPT_ITA**
D.T.68.	An outsider may gain the ability to observe TSF data while the data is being transferred between TSFs.	**FPT_ITC**
D.T.69.	A user may gain access to TSF data or to TSF protected resources that results in undetected and unauthorised modification of TSF data.	**FPT_ITI**
D.T.70.	A human user may physically access the TSF that results in the undetected use, modification, substitution, or analysis of the physical representation of the TSF.	**FPT_PHP**
D.T.71.	The TSF may enter an insecure state after completion of system initialisation due to the failure of some TOE component.	**FPT_RCV**
D.T.72.	An untrusted subject may violate the TSP as a result of the subject bypassing the TSF enforcement mechanism and directly accessing TOE data and or TOE resources.	**FPT_RVM**
D.T.73.	An unauthorised observation of the TSF's internal code or data may result due to the lack of domain separation functions.	**FPT_SEP**
D.T.74.	An unauthorised modification of the TSF's internal code or data may result due to the lack of domain separation functions.	**FPT_SEP**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.75.	An unauthorised observation and/or modification of a subject's address spaces may result due to the lack of domain separation functions.	**FPT_SEP**
D.T.76.	A non-policy enforcing mechanisms in the TSF may interfere with the operation of the policy enforcement mechanisms due to the lack of domain separation functions.	**FPT_SEP**
D.T.77.	The TSF may inadvertently modify itself due to TSF design errors.	**FPT_SWM**
D.T.78.	The TSF may be used in an inappropriate manner by entities external to the TOE upon which the TOE's security depends (e.g., administrators, operators).	**FPT_TSA**
D.T.79.	The TSF may fail to meet its security objectives due to the lack of the TOE to provide proper administrative functions.	**FPT_TSM**
D.T.80.	The TSF may fail to properly respond to external disasters due to the lack of the TOE to provide proper backup and restoration capabilities.	**FPT_TSM**
D.T.81.	An internal failure of the TSF may go undetected resulting in a TSP violation.	**FPT_TST**
D.T.82.	The internal inconsistency of the TOE may occur and remain undetected resulting in a TSP violation.	**FPT_TST**
D.T.83.	An unskilled authorised administrator may use a TSF interface in an insecure manner.	**FPT_TSU**
D.T.84.	In the event of a hardware failure the TSF may enter a state that is not consistent with the TSP.	**FRU_FTL**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.85.	A denial of service may result due to an authorized user consuming a disproportionate share of resources.	**FRU_RSA**
D.T.86.	A denial or inappropriate delay of service may result due to a low priority task interrupting, or delaying, a high priority task.	**FRU_PRS**
D.T.87.	A user or entity may send data to another user or subject and subsequently deny the act of ever sending the data.	**FCO_NRO**
D.T.88.	A user or entity may receive data from another user or subject and subsequently deny the act of ever receiving the data.	**FCO_NRR**
D.T.89.	The TSF may not be able to protect user data in a manner that is consistent with the expectation of its “owners,” due to the inability of the TSF to instantiate the user’s protection needs.	**FDP_ACC**
D.T.90.	The TSF may not be able to protect user data in a manner that is consistent with an organisational security policy, due to the inability of the TSF to instantiate the policy.	**FDP_ACC**
D.T.91.	The TSF may not be able to correctly associate the sensitivity of an object with the objects attributes, due to the inability of the TSF to initialise object attribute values.	**FDP_AIC**
D.T.92.	The TSF may not be able to protect user data in a manner that is consistent with an organisational security policy, due to the inability of the TSF to enforce the policy.	**FDP_ACF**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.93.	Exported user data may not be properly protected under the control of a remote TSF due to the inability of the remote TSF to properly associate security attributes with the exported data.	**FDP_ACF**
D.T.94.	User data may be exported in a manner that is in violation of the TOE's export protection policy due to the inability of the TOE's TSF to properly associate security attribute(s) with the user data during data export.	**FDP_ETC**
D.T.95.	User data may be exposed to a user that is not authorized to view the information due to the inability of the TSF to properly instantiate an information flow control policy over the set of TOE operations that are capable of acting on the user data.	**FDP_IFC**
D.T.96.	User data may be modified by a user that is not authorized to alter the data due to the inability of the TSF to properly instantiate an information flow control policy over the set of TOE operations that are capable of acting on the user data.	**FDP_IFC**
D.T.97.	User data may be exposed to a user that is not authorized to view the information due to the inability of the TSF to properly enforce an information flow control policy over the set of TOE operations that are capable of acting on the user data.	**FDP_IFF**
D.T.98.	User data may be modified by a user that is not authorized to alter the data due to the inability of the TSF to properly enforce an information flow control policy over the set of TOE operations that are capable of acting on the user data.	**FDP_IFF**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.99.	User data that is imported from outside of the TSC by the TSF may not be able to properly determine the security attributes of the imported data.	**FDP_ITC**
D.T.100.	A user or subject may gain access to information that is in violation of the TSP due to the inability of the TSF to remove residual data from public objects after prior usage.	**FDP_RIP**
D.T.101.	The integrity of a TOE database may be lost due to the inability to rollback the database to a previously known state in the event that a transaction or processing function failed to complete.	**FDP_ROL**
D.T.102.	The values of security attributes may be modified by unauthorised users, due to the inability to control access to these attributes.	**FDP_SAM**
D.T.103.	The TOE's user data protection policy may not be able to be properly maintained due to the inability of the TSF to provide the capability for an authorised administrators to modify data protection relevant security attributes.	**FDP_SAM**
D.T.104.	The TOE's user data protection policy may not be able to be properly maintained due to the inability of the TSF to provide the capability for an authorised user to retrieve relevant security attributes.	**FDP_SAQ**
D.T.105.	The confidentiality of user data may be lost due to the inability to protect the data during data exchange.	**FDP_UCT**
D.T.106.	The integrity of user data may be lost due to the inability to protect the data during data exchange.	**FDP_UIT**

Table B.3 - Detailed Threats

D.T. #	Detailed Threat	Related Family
D.T.107.	A security relevant event may occur and be recorded, without the user that was responsible for the event being held accountable for the event.	**FAU_GEN.2**
D.T.108.	An application may provide a user with information that is perceived by the user as coming from the TSF.	**FTP_TRP, FTP_ITC**
D.T.109.	An application may intercept user private data (e.g., password) and subsequently use that data to impersonate the user.	**FTP_TRP, FTP_ITC**
D.T.110.	A user observes the identity of another user.	FPR_ANO, FPR_PSE
D.T.111.	A user may violate the TSP and not be accountable due to the protection of the user identity.	**FPR_PSE**
D.T.112.	A user may determine a user's identity or other privacy sensitive information through observing an aggregate of events related to the user in the system.	**FPR_UNL**
D.T.113.	A user may obtain relevant information about other users by observing the use of a resource.	**FPR_UNO**

Table B.3 - Detailed Threats

Annex C

CC observation report (CCOR)

C.1 Introduction

1060 The CC sponsoring organisations welcome feedback from the community and are particularly interested in observations and comments arising out of trial application of the criteria.

1061 The CC sponsoring organisations have set up a body, the Common Criteria Implementation Board (CCIB), to coordinate and learn from the community experience and to ensure that future issues of the CC can benefit from that experience.

1062 Comments, observations, and requests for interpretations should be sent to one of the addresses listed inside the front cover of the CC. If you require feedback on a specific evaluation matter, you should use the contact address which corresponds to the evaluation authority concerned.

C.2 Categorisation of observation report

1063 In order to allow automated categorisation of the observations, a standard observation format is needed. Each observation should include an identifier as to whether the comment pertains to the ***approach*** in the CC, the technical ***detail*** of any specific portion of the CC, or ***editorial*** work that needs to be done. Additionally, for comments on technical detail, an indication of the scope of the comment (e.g., ***local***, ***global***) should be provided.

1064 The following provides a description of each of these terms:

a) *Approach*: observations requesting further guidance relating to the approach of the CC which the author of the observation report considers to be fundamental to the further progress of the CC or trial application of the criteria should be marked with this identifier.

b) *Detail*: Specific observations on technical details of the CC should be marked with this identifier. These comments should be further categorised as either local or global.

Local: is applicable to a single specific class, family, component, or element.

Global: is applicable to multiple classes, families, components, or elements.

c) *Editorial*: typographical and grammatical errors, as well as comments on presentation style.

Local: is applicable to a single specific class, family, component, or element.

Global: is applicable to multiple classes, families, components, or elements.

C.3 Format of observation report

1065 The following provides a description of each of the structure of the required comment format and an example of a comment in the required format.

1066 If you are submitting one or more observations by electronic mail or other machine readable format, please insert the tags defined below starting in the first column as this will greatly assist in any automated handling of your input.

1067 Each observation report should consist of three parts.

a) The first part consists of a tags **$1:** to **$4:**, which includes the information to allow the unique identification of the originator. This first set of tags is required only once per single observation or batch of observations.

b) The second part consists of tags **$5**: to **$9:**, which includes the information to allow the unique identification and categorisation of the observation, the actual observation itself and suggested solution. The text of each observation should extend to as many lines as are needed to fully express the observation. There can be one or more observations in an observation report.

The set of tags **$5**: to **$9:**, comprising this second part of the observation report, should be repeated for each observation being submitted.

c) The third part consists of a single terminating tag **$$:**. This final tag is required only once per single observation or batch of observations.

C.3.1 Tag definitions for observation report

$1: Originator name

1068 Name of commenter (only required once per message).

$2: Originator organisation

1069 Originator organisation/affiliation (only required once per message).

$3: Return address

1070 Electronic mail or other address for response (only required once per message).

$4: Date

1071 Submission date of observation YY/MM/DD (only required once per message).

$5: Originator report reference identification

1072 Reference for observation which is unique to originator. Please include your initials or similar unique discriminator, e.g., ABC1234.

$6: One line summary/title of observation

1073 Short summary/title for problem (up to 60 characters).

$7: CC document reference

1074 Single reference to the affected area of the CC as detailed as appropriate. Where possible, part number, section, paragraph, class, family, component, or requirement reference should be provided.

1075 The template for CC document reference is as follows:

$7: Part / Section / Paragraph / [Approach / Detail - [Local / Global] / Editorial] - [Local / Global] / [Keyword]

1076 The CC document reference template should be completed as follows (see below for completed example):

a) The characters "**$7:**", to indicate the start of an observation.

b) Identification of the CC part, section, and paragraph to which the comment applies in the CC. All 3 pieces of identifying information should be provided, each separated by a slash character (/).

Valid identifiers for the CC Part are e.g., part 1 or 1, part 2 or 2, part 3 or 3, and profiles or PP.

Identification for the CC section should be either a section number (e.g., 1.3.2), if applicable, or, for requirement classes, families, or components, the name of the class (e.g., FIA), family (e.g., FIA_ATD), or component (e.g., FIA_ATD.1).

c) Identification of the reviewer's categorisation of the observation. Brackets "[..]" indicate that the reviewer should choose *one* of the options contained within the brackets, these can be abbreviated to the initial character only (e.g., "A" , "D - L", or "E - G").

d) An optional keyword.

1077 Any identification field should be left blank or be filled with an asterisk (*) to indicate that the field is not applicable or necessary for the comment.

$8: Statement of observation

1078 Comprehensive statement of observation or query, contains the actual text of the observation. Should include specific reference to examples of the observation, where appropriate.

$9: Suggested solution

1079 Proposed solution or solution approach.

$$: Terminating tag.

1080 This enables any automated handling to determine the end of the batch of observations (only required once per batch of observations).

C.3.2 Example observations:

$1: A. N. Other

$2: PPs ‘R’ US

$3: another@ppsrus.com

$4: 960131

$5: ano.comment.1

$6: Presentation comment.

$7: 1 / 8.1 / 90 / Editorial - Local /

$8: The word “global” at the end of the first line should be italicised.

$9: Italicise “global”.

$5: ano.comment.2

$6: Missing requirement for audit.

$7: 2 / FAU / 336 / Detail - Local /

$8: The first sentence of this paragraph is incomplete.

$9: The first sentence should include “imminent” violations.

$5: ano.comment.3

$6: Problems in navigating the document.

$7: 2 / * / * / Approach / threats

$8: The statements of threat in the functional families are largely re-statements of the family behaviour from the threat viewpoint. Does this material need to be re-stated twice within the functional families?

$9: Could all threat information be described in a separate section with a table mapping the various functional components to the threats they address?

$$: This is the end tag, the contents are immaterial.

C.4 Printed observation report

1081 An example of a printed observation report is provided in Table C.1.

COMMON CRITERIA OBSERVATION REPORT	
$1:	**Originator Name**
$2:	**Originator organisation**
$3:	**Return address**
$4:	**Date**
$5:	**Originator report reference identification**
$6:	**One line summary/title of observation**
$7:	**CC document reference**
$8:	**Statement of observation**
$9:	**Suggested solution**
$$:	

Table C.1 - CC observation report